

Universidad del Valle
Facultad de Ingeniería
Escuela de Ingeniería en Sistemas y Computación



Refinamiento de la planificación eco-eficiente de estiba en
grandes buques portacontenedores.

Trabajo de grado presentado por
Adolfo León Canizales Murcia

Santiago de Cali
Junio 2014

Universidad del Valle
Facultad de Ingeniería
Escuela de Ingeniería en Sistemas y Computación



Refinamiento de la planificación eco-eficiente de estiba en
grandes buques portacontenedores.

Estudiante: Adolfo León Canizales Murcia
leon99adolfo@gmail.com

Director: Alberto Delgado, Ph.D.
Optivation Aps

Codirector: Juan Francisco Díaz Frías, Ph.D.
Profesor Titular
Universidad del Valle

Santiago de Cali
Junio 2014

Índice general

Índice general	I
1 Contexto y objetivos	5
1.1. Desarrollo e investigación	5
1.1.1. Planteamiento del problema	5
1.2. Objetivos	7
1.2.1. Objetivo general	7
1.2.2. Objetivos específicos	7
1.2.3. Productos esperados	7
1.3. Justificación	7
1.3.1. Justificación económica	8
1.3.2. Justificación social	8
1.3.3. Justificación académica	8
2 Marco Teórico	11
2.1. Contenedores	11
2.2. Buques porta-contenedores	12
2.3. Terminales de contenedores marítimos	13
2.4. Proceso de estiba y trincaje	13
2.4.1. Estiba de contenedores	13
2.4.2. Trincaje de contenedores	15
2.4.3. Objetivos de los procesos de estiba y trincaje	15
2.4.4. Planes de estiba	15
2.5. CP - Programación por restricciones	16
2.6. C++	17
2.7. Gecode	18
3 Estado de arte	19
3.1. Métodos basados en modelos únicos	19

3.2. Métodos basados en la descomposición del problema en varios modelos de optimización	20
3.3. Descripción del modelo en restricciones de Delgado et al (2012)[8].	21
4 Descripción del problema	27
4.1. Reglas y límites del problema de estiba	27
4.2. Proceso de estiba y trincaje en la computación	28
5 Introducción a Modelos	31
5.1. Abstracciones de restricciones	32
5.2. Conjunto de constantes	35
6 Modelo original	39
6.1. Variables de decisión	39
6.2. Restricciones del modelo	39
6.3. Estrategia de distribución	46
6.3.1. Estrategia de distribución por pilas específicas	47
6.3.2. Estrategia de distribución por pilas general	48
6.3.3. Estrategia de distribución por niveles de cada pila	48
7 Modelo extendido	51
7.1. Conjunto de constantes adicionales	51
7.2. Variables de decisión	51
7.2.1. Restricciones del modelo	51
7.3. Estrategia de distribución	56
7.3.1. Estrategia de distribución por pila específica	56
7.3.2. Estrategia de distribución por niveles de cada pila	56
7.3.3. Estrategia de distribución por niveles de cada pila y por contenedores reales	56
8 Implementación de modelos.	59
9 Resultados de los modelos	63
9.1. Clasificación de instancias a probar	63
9.2. Resultados de los modelos	64
10 Conclusiones	71
Lista de Símbolos y Abreviaciones	73
Índice de figuras	74
Índice de cuadros	76

Agradecimientos

Quiero agradecer a mi padre y a mi madre, Adolfo León y María Patricia, por brindarme su apoyo, compañía y eterno amor. A mis hermanas, Diana Fernanda y María de los Ángeles, por ser las mujeres con las que siempre contaré. A todos mis familiares por estar incondicionalmente disponibles para mí. Igualmente, agradezco a mi tutor Alberto Delgado y al mi profesor Juan Francisco Díaz que me aportaron los conocimientos que hacen posible que logre una meta más en mi vida. A Carolina la gran mujer que me acompaña en mis dichas y desdichas, a mis fieles amigos que siempre tendré en el corazón y a todas las personas que me han apoyado y me hacer crecer personal y profesionalmente. Sinceramente, la fortuna me sonrió al poder dedicar éste trabajo a todas aquellas personas que hacen parte de mi vida. Muchas gracias a todos.

Resumen

Generar planes de estiba en buques portacontenedores es un problema de optimización combinatorio que afecta en gran medida la economía marítima mundial. El proyecto expuesto en éste documento hace parte de una descomposición heurística del problema de generar planes de estiba. La descomposición separa el problema en dos fases generales denominadas *planificación maestra*, la cual distribuye grupos de contenedores en áreas de almacenamiento del buque y *planificación de ranura* que tiene como objetivo asignar los grupos de contenedores, clasificados en la primera fase, de forma individual en sus respectivas áreas de almacenamiento. Este proyecto se enfoca exclusivamente en encontrar soluciones factibles, utilizando un modelo en restricciones, a la *planificación de ranura* ubicada sobre la cubierta del buque portacontenedores. Este problema es NP-Completo, sin embargo, se encuentra solución al 99 % de las 234 pruebas reales ejecutadas en pocos segundos, y donde el 61 % de las soluciones, son las más óptimas posibles. Optimización, Programación por restricciones, Planificación de estiba en buques portacontenedores, Planificación de ranura.

Capítulo 1

Contexto y objetivos

1.1. Desarrollo e investigación

1.1.1. Planteamiento del problema

En la sociedad, a lo largo de los últimos tiempos, el transporte masivo de productos por vía marítima ha sido de gran importancia, debido a que la movilización de productos entre diferentes regiones cumple un factor de progreso, mejoramiento y bienestar. La industria del transporte marítimo de mercancías posee un gran impacto en la economía mundial. Este acto se ve reflejado en el incremento considerable y evolución de la flota mundial de buques mercantes y puertos, con el transcurso de los años.

El transporte por mar de los diferentes tipos de mercancía es realizado por diversas clases de buques, lo anterior se da, en función de cómo es presentada esta mercancía para su embarque. Debido a notables ventajas con relación a la rapidez, eficacia y disminución de pérdidas por avería, uno de los más importantes embarques para carga seca son los buques portacontenedores. Estos buques están diseñados para estibar y transportar diferentes tipos de mercancía, inclusive mercancía refrigerada.

Actualmente, los buques portacontenedores son el medio más utilizado para las tareas de transporte de productos entre diferentes regiones. Estos buques satisfacen gran porcentaje, el 90 %, de la movilización de productos en todo el mundo y es difícil imaginarse otro tipo de transporte que satisfaga dicha necesidad.

Como se puede notar, es de gran importancia que el proceso de transporte de productos en los buques portacontenedores sea rápido y efectivo. Sin embargo, este proceso de transporte posee el problema del cómo transportar la mercancía cumpliendo determinadas características. Como respuesta a este problema, surge la construcción de herramientas que implementen modelos de optimización para mejorar el proceso de embarque y desembarque de contenedores en un barco de una forma refinada y

eficaz.

La optimización de procesos de estiba en los puertos costeros es un proceso que implica encontrar una forma conveniente de organizar un conjunto de cajas metálicas, llamadas contenedores, en un buque. Este ordenamiento debe estar enfocado en la búsqueda de la consecución de algunos objetivos, como por ejemplo la seguridad del barco, es decir, que el barco pueda soportar los viajes en alta mar sin que éste sufra daños o se hunda. Adicionalmente, se tendría en cuenta la rapidez para la carga y descarga de contenedores del buque, en el menor tiempo posible, en cada uno de los puertos que atraca.

De esta manera, uno de los más fuertes fundamentos del sector portuario es una buena planeación de estiba, la cual se ha estado realizando por coordinadores que utilizan herramientas gráficas que les permite generar un plan para la embarcación. Sin embargo, debido a la complejidad del problema, se ha incrementado el interés por la ampliación de estas herramientas, que fusionadas con algoritmos veloces de optimización, podrían ser de gran ayuda en la planeación de estiba.

En la actualidad, existen diversas maneras de planear la estiba de contenedores en estos tipos de buques. Estos planes están dados por herramientas gráficas y recientemente, algoritmos de optimización que pretenden encontrar la mejor manera de organizar los contenedores en los buques, teniendo en cuenta factores como espacio, tiempo y seguridad en su transporte.

En este proyecto, se busca ampliar la planeación de estiba por medio de modelos de algoritmos de optimización a la carga localizada sobre la cubierta de los buques portacontenedores. Lo anterior, con el fin de mejorar el modelo presentado por Delgado et al(2012) [8] que hace parte de la segunda fase de un planteamiento propuesto para estibar contenedores en un buque.

El enfoque utilizado para resolver el problema de estiba consta de dos partes. La primera parte es un modelo de programación entera usada para la asignación de grupos de contenedores en las áreas de almacenamiento de la embarcación a través de múltiples puertos [14] y la segunda parte se enfoca en un modelo de restricciones para estibar contenedores de forma individual en las locaciones de un buque y así poder estibar el buque en su totalidad [8]. Esto es posible gracias a que en la primera parte ya se ha descompuesto el problema (Ver Figura 1.1).

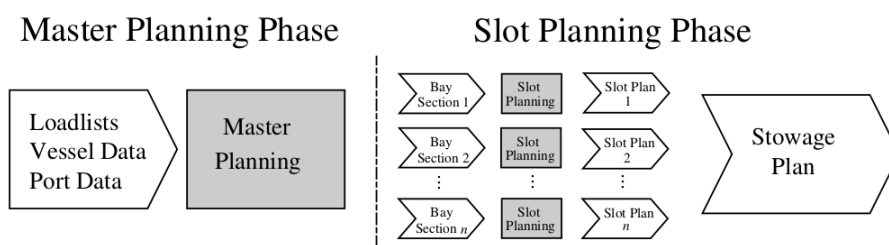


Figura 1.1: Modelo total de planeación de estiba. [14][8]

1.2. Objetivos

1.2.1. Objetivo general

Extender y refinar el modelo computacional presentado por Delgado et al (2012) [8].

1.2.2. Objetivos específicos

- Extender el modelo computacional de estibación teniendo en cuenta los siguientes aspectos:
 1. No provocar la desestabilización del buque.
 2. Contemplar las características principales del trincaje que se realiza en los contenedores.
 3. Minimizar el número de contenedores que no se cargan en el buque.
- Implementar el modelo en GECODE.[10]
- Ejecutar y documentar un plan de pruebas del nuevo modelo computacional para estibación de contenedores.

1.2.3. Productos esperados

Objetivos Específicos	Producto(s) Esperados
Extender modelo computacional.	Modelo extendido que permita nuevas formas de estibación, intentando implementar características deseables, teniendo en cuenta el trincaje y la no desestabilización del buque.
Implementar modelo en GECODE.	Código que permita mostrar y corroborar el funcionamiento del modelo realizado.
Plan de pruebas.	Desarrollo y documentación de las pruebas a realizar del nuevo modelo, con datos reales.

1.3. Justificación

El refinamiento y optimización de modelos encierra varios aspectos interesantes. La idea de representar de manera computacional una buena manera de estibar en los barcos portacontenedores es una buena perspectiva desde varios puntos de vista. El poder modelar una herramienta capaz de ordenar contenedores con ciertas reglas o restricciones, hace posible que los partícipes encargados del proceso tengan una visión

más amplia de conocimiento, que les puede otorgar una gran ayuda, al momento incurrir en la acción de estibar contenedores.

1.3.1. Justificación económica

La industria del transporte marítimo de mercancías posee un gran impacto en la economía mundial, ya que la mayoría de las mercancías es transportada en barcos portacontenedores. Por lo anterior, se puede afirmar que aunque las mejoras en la optimización del proceso de estiba sean pequeñas, éstas representan un gran impacto a nivel económico; la generación de planes óptimos de estiba representa grandes ganancias económicas tanto para los barcos como para los puertos.

Por una parte, tenemos que la generación de un buen plan de estiba permite a los buques una menor estadía en el puerto en el cual se encuentre anclado. Este punto es importante, ya que el tiempo de estadía de un barco en el puerto es un cargo económico que el barco debe pagar; la minimización de este tiempo reduce los costos de cargar y descargar contenedores en el puerto. Adicionalmente, la reducción del tiempo en el puerto se puede utilizar en la disminución de la velocidad del barco para navegar en alta mar de un puerto a otro, lo que rebaja considerablemente la cantidad de combustible que éste debe utilizar.

Por otra parte, los buenos planes de estiba permiten a los puertos atender una mayor cantidad de buques en determinados periodos de tiempo. En esta industria, atender varios buques es más satisfactorio, económicamente, que cobrar la estadía de unos cuantos buques.

1.3.2. Justificación social

El impacto social está enfocado hacia la comunidad que se desempeña o estudia en el área relacionada con las actividades de estibación en los puertos costeros. Las personas en este mercado obtendrían la posibilidad de encontrarse con un modelo y una herramienta que los ayudará a generar una idea más general del problema de ordenamiento de contenedores, tanto en el buque, como en el puerto. Además, este proyecto también posee impactos ecológicos, debido a que un buen plan de estiba permite que los buques no estén mucho tiempo en el puerto y abre la posibilidad de que los barcos naveguen más lentamente en alta mar hacia su nuevo puerto, permitiendo un menor consumo de combustible, y por ende, una menor producción de CO_2 .

1.3.3. Justificación académica

Con este proyecto se pretende la extensión y el refinamiento de un modelo de estibación, lo que le permitirá a las personas que se desenvuelvan en estos ámbitos, estudiar y analizar los procesos de ordenamiento de contenedores en los buques y en los puertos. Como consecuencia, estas personas podrán entender y responder a las

exigencias que esta área impone a la hora de incursionar en este mercado. El nuevo modelo estará en la capacidad de proporcionar los conocimientos claves del mercado costero.

Capítulo 2

Marco Teórico

2.1. Contenedores

Los contenedores son cajas de metal donde se puede almacenar muchos tipos de bienes. Los contenedores fueron diseñados para facilitar su movilidad sobre diferentes tipos de transportes como son los trenes, buques, camiones, entre otros. En promedio, los contenedores poseen un tiempo de vida cercano a los 15 años, esto es debido a que su estructura les permite ser reutilizados un gran número de veces. Existen diferentes tipos de contenedores y éstos son utilizados dependiendo de los requisitos específicos de la carga. Normalmente, los contenedores tienen un ancho de 8 pies, un alto de 8.6 (altura estándar) o 9.6 (cúbico) pies y un largo de 20 o 40 pies, los contenedores con estas características representan en promedio el 90 % del inventario de contenedores que se encuentran en circulación por el mundo (Ver figura 2.1). [7]

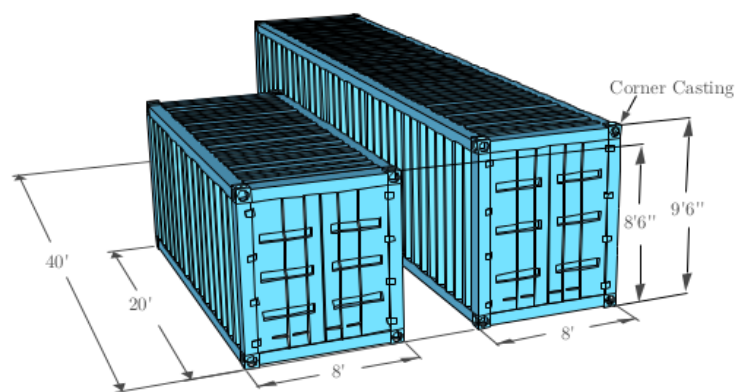


Figura 2.1: Dimensiones comunes de un contenedor.

También se encuentran diferentes tipos de contenedores, como son los contenedo-

res estándar de 10, 45, 48 y 53 pies de largo. Existen otros tipos de contenedores que son utilizados para transportar carga que necesitan una ambientación especial. Por ejemplo, existen cargas que necesitan una constante ventilación, productos orgánicos, que es proveída por contenedores ventilados que poseen pequeñas aberturas las cuales permiten la constante circulación del aire. Por otro lado, están las carga que son sensitivas a la temperatura y necesitan mantener un punto de congelación para prevenir daños en la carga, como por ejemplo la comida congelada. Para este tipo de carga se utilizan los contenedores refrigerados. Los contenedores refrigerados poseen un motor de enfriamiento eléctrico, y necesitan ser almacenados y conectados a una fuente de poder eléctrica. Por otro lado, existen otros tipos de contenedores como son los que transportan mercancías peligrosa y que necesitan un grupo específico de reglas para ser transportados.

2.2. Buques porta-contenedores

Los buques portacontenedores son barcos celulares diseñados y construidos con el único propósito de transportar contenedores a través del mar. La estructura de un buque portacontenedores esta dividida longitudinalmente por secciones llamadas bahías en las que se almacenan los contenedores. Las bahías se encuentran distribuidas en todo el barco, sobre y bajo cubierta, separadas por tapas de escotilla a lo largo del buque. Estas bahías se separan transversalmente en pilas que tienen un ancho de 8 pies y un largo de 20 (TEU - veinte pies unidad equivalente), 40 (FEU - cuarenta pies unidad equivalente) o 45 pies en algunos casos. Un grupo de pilas, sobre o debajo de cubierta, que compartan la misma puerta de escotilla es llamada una localidad. Adicionalmente, cada pila se encuentra dividida en celdas, las cuales se encuentran organizadas verticalmente e indexadas por niveles. Cada celda se divide en 2 ranuras que separan la celda en popa y proa. Adicionalmente, algunas ranuras de las celdas posee un enchufe que proporciona energía eléctrica a contenedores que necesitan refrigeración (ver Figura 2.2 y 2.3). [8]

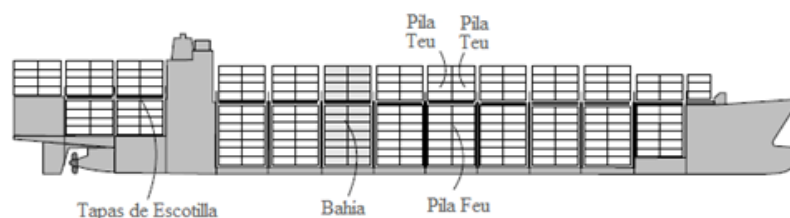


Figura 2.2: Imagen de un buque portacontenedores.

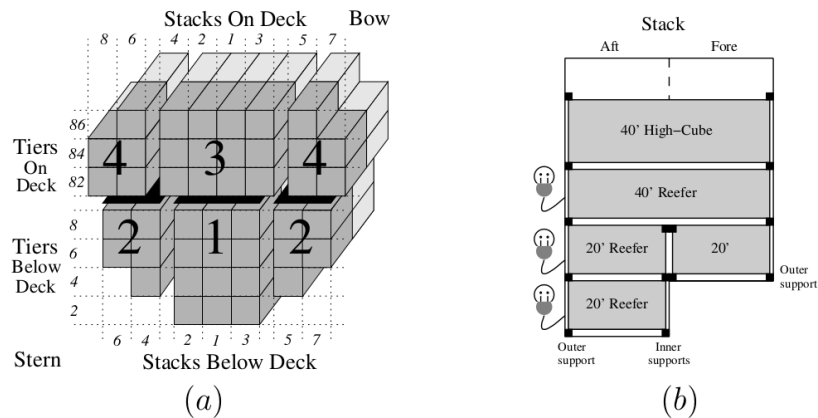


Figura 2.3: (a) Vista frontal de una bahía. (b) Vista lateral de una pila con carga parcial. Cada enchufe de poder indica si puede ir un contenedor refrigerado.

2.3. Terminales de contenedores marítimos

Las terminales de contenedores marítimos son secciones del puerto donde los contenedores son cargados desde un vehículo terrestre a un buque portacontenedores o viceversa. Estas terminales se encuentran divididas en partes principales, estas son:

- El muelle. Esta es la sección donde los buques atracan y donde los contenedores son cargados y descargados.
- El patio. Esta sección tiene como objetivo almacenar temporalmente los contenedores en tierra.
- La playa de contenedores. En esta sección es donde los contenedores son recogidos o dejados por trenes o camiones.

2.4. Proceso de estiba y trincaje

El proceso de estiba y trincaje se fundamenta en el hecho de que los buques portacontenedores viajan por el mundo cargando y descargando contenedores en cada puerto costero en el que atracan. A continuación, se presenta una breve explicación de estos dos procesos:

2.4.1. Estiba de contenedores

Se define como estiba a la técnica de colocar la carga a bordo de un buque para ser transportada con un máximo de seguridad, tanto para el buque, como para su tripulación, ocupando el mínimo espacio posible, evitando averías en el procedimiento y reduciendo al mínimo las demoras en el puerto de descarga[18]. La estiba de

contenedores no es un proceso trivial debido a que los contenedores no se pueden almacenar libremente en el barco. Lo anterior se debe, a que el almacenamiento de contenedores debe cumplir con un grupo de reglas obligatorias que previenen futuros daños o desastres.

Tipos de estibación en buques portacontenedores

Existen dos procedimientos bien diferenciados en la estiba de contenedores en los buques portacontenedores. La información sobre estos tipos de procedimiento fue obtenida de un documento en la Web [6]:

- Mediante guías celulares en el buque. Las guías son un conjunto de raíles metálicos que se ubican de manera vertical y que forman celdas del tamaño del contenedor, 20 x 8 pies o de 40 x 8 pies. Estas guías permiten apilar en cada celda, sin necesidad de realizar un trincado sobre los contenedores (Ver 2.4.2). Adicionalmente, La limitación en la altura de una pila de contenedores la establece la resistencia del contenedor más bajo, es decir, el contenedor mas abajo de la pila debe soprtar la carga de todos los contenedores apilados sobre él. Este es el procedimiento de estiba en bodegas de los buques dedicados exclusivamente al transporte de contenedores; para la estiba sobre cubierta y sobre tapas de escotilla se han buscado soluciones similares, e incluso se han construido buques con bodegas abiertas, “hatchless”, para obviar el problema. Este tipo de estiba no es muy utilizado debido a que las guías celulares implican un gran costo para el buque.
- Sin guías celulares en el buque. En este caso, los contenedores se colocan agrupados en bloques, trincados para garantizar la resistencia de cada bloque a los movimientos del buque. Este procedimiento de estiba es mucho más laborioso que el de las guías celulares, y es típico de los buques polivalentes que transportan contenedores. Este procedimiento se ha usado frecuentemente sobre las escotillas en los porta-contenedores celulares.

En la estiba sin guías celulares se utiliza con mucha frecuencia el trincaje para el aseguramiento de los contenedores. El proceso de trincaje de contenedores tiene como objetivo evitar el daño de los contenedores cuando éstos se encuentra expuestos a movimientos bruscos, que el buque experimenta. Estos movimientos se deben a la inestabilidad de la superficie del mar, al viento y a la estabilidad del buque. Los barcos se balancean, cabecean, suben y bajan bruscamente, realizan pantocazos (golpes del casco contra las olas) y guiñadas (rotación del buque alrededor de sus ejes verticales) y estos movimientos hacen que los contenedores cargados sean sometidos a fuerzas, principalmente en dirección oblicua y vertical.

2.4.2. Trincaje de contenedores

El proceso de trincaje se encuentra estrechamente ligado con el proceso de estiba de contenedores. Lo anterior, debido a que el trincaje es utilizado para asegurar los contenedores al buque, una vez los contenedores han sido estibados. Normalmente, el proceso de trincaje se aplica sobre los contenedores que se encuentran por encima de la cubierta del barco, ya que estos contenedores se encuentran expuestos a una mayor cantidad de fuerzas externas.

Las trincas que sujetan los contenedores al barco pueden ser de barras, cables o cadenas. Las pilas de los contenedores estibados se amarran con estas trincas y van desde la cubierta hasta las cantoneras inferiores de los contenedores que se estiban encima de otro. Estas trincas tienen comúnmente como máximo las cantoneras inferiores del tercer contenedor que se encuentre apilado. Por lo cual, las cargas estibadas por encima de estos contenedores deben ser más livianas.

Una característica que es utilizada para facilitar el proceso de trincaje, es el cálculo del centro de gravedad de las pilas estibadas en el barco. El proceso de trincaje se facilita cuando el centro de gravedad en el eje *Y* se encuentra a un tercio de la altura de la pila.

2.4.3. Objetivos de los procesos de estiba y trincaje

En el proceso de estiba y trincaje se debe tener en cuenta las siguientes metas u objetivos.

- La seguridad del buque y sus tripulantes.
- La minimización en el tiempo de carga y descarga de contenedores en el buque.
- Cargar la mayor cantidad de contenedores al buque.
- La estabilidad del buque.
- La integridad de la carga transportada.
- La minimización del uso de trincas.
- Reducir impacto ecológico.

2.4.4. Planes de estiba

Los planes de estiba son los que definen como se procederá y como se asignarán los contenedores en el buque. Los planes de estiba son creados por coordinadores encargados de cumplir con la tarea de cargar los contenedores en el buque antes de que éste atraque en el puerto. Para la generación de planes de estiba los coordinadores tienen la lista de contenedores a cargar en el buque. Esta información incluye detalladamente las características de los contenedores tales como el peso, el alto, el largo,

el tipo de contenedor y el puerto de descarga que tiene como destino. Los coordinadores también tienen información acerca de la estructura del buque, del combustible y de los contenedores ya cargados. Adicionalmente, el coordinador posee información del puerto actual y de los puertos que le prosiguen. Esta información incluye la profundidad del muelle y la cantidad de grúas habilitadas para cargar los contenedores. El plan de estiba debe cumplir con todas las reglas y límites de estiba (Sección 4). Además, este plan debe tener como prioridad reducir el costo total, sin afectar la seguridad en todo el proceso. Existen dos motivos que impactan circunstancialmente el tiempo que un buque permanece en el puerto. El primero, radica en la cantidad de movimientos que tiene que hacer la grúa para ordenar de manera adecuada los contenedores que ya están en el buque. El segundo, se presenta por la cantidad de movimientos innecesarios que se hacen para descargar contenedores del buque. (Ver figura 2.4)

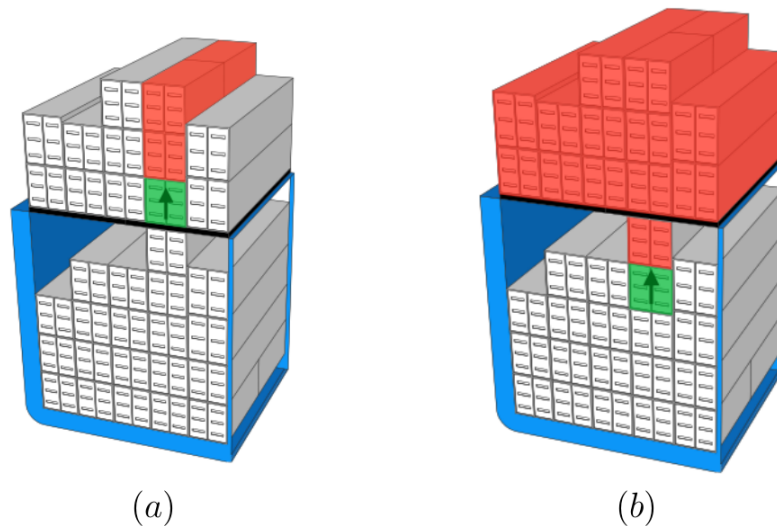


Figura 2.4: Los contenedores de color verde se descargarán antes que los rojos. (a) Pila sobre-estibada . (b) Escotilla sobre-estibada.

Al final, el plan debe contener la lista de contenedores con la posición exacta donde cada uno de ellos será estibado.

2.5. CP - Programación por restricciones

La programación por restricciones consiste en un conjunto de técnicas para resolver problemas de satisfacción de restricciones. Un problema de satisfacción de restricciones (CSP), es un conjunto de restricciones aplicadas sobre un conjunto de variables. Una restricción, en este contexto, es simplemente una relación lógica, por ejemplo, tenemos un par de enteros X y Y y decimos que " X es menor que Y ".

En la programación por restricciones se busca resolver dos problemas. El primero es decidir si existe una solución, sin tener que construirla obligatoriamente y el segundo problema es encontrar una o más soluciones.

Ideas básicas de la programación por restricciones

La programación por restricciones esta básicamente separada en dos pasos: la propagación y la búsqueda. Estos dos pasos se enfocan en tres ideas importantes, que son las siguientes:

1. Guardar información parcial de la solución. Durante el calculo, se puede tener información parcial de las variables que conforma una solución. Por ejemplo, en cualquier solución al problema sabemos que " X es mayor 100".
2. Usar deducción local. La deducción local se puede dar cuando combinamos diferentes restricciones o cruzamos la información parcial que tenemos. Por ejemplo, si combinamos " X es menor que Y " con " X es mayor 100" podríamos deducir que " Y es mayor 101".
3. Se realiza búsqueda controlada. Cuando con la información parcial no se pueden hacer mas deducciones locales, se pasa a realizar la denominada búsqueda. La idea es buscar la solución o las soluciones lo menos posible. Se realiza una etapa de búsqueda y se deduce nuevamente. En una etapa de búsqueda lo que se hace es dividir el problema P en 2 partes $(P \wedge C)$ y $(P \wedge \neg C)$, donde C es una nueva restricción y a partir de allí se realizan nuevas deducciones y así se continua hasta encontrar todas las soluciones. La escogencia de C es muy importante, debido a que una restricción C bien escogida llevará a las soluciones en pocas etapas de búsqueda.

2.6. C++

Es un lenguaje de programación que se cataloga como una evolución del lenguaje C. El lenguaje fue creado en el año 1980 por Bjarne Stroustrup y están son algunas de sus características [15]:

- Lenguaje compilado: El lenguaje C++ compila directamente el código nativo sobre la maquina. Lo anterior, permite que sea uno de los lenguajes mas rápidos del mundo.
- Lenguaje abierto estandarizado por la ISO: Desde el año 1998 el lenguaje C++ fue estandarizado por un comité de la ISO.
- Lenguaje inseguro inflexible de tipos: Se espera que la persona que maneje el lenguaje sepa lo que esta haciendo. Por lo anterior, el lenguaje brinda una gran cantidad de controles como resultado.

- Varios paradigmas: El lenguaje permite trabajar con diferentes tipos de paradigmas, tales como el procedimental, el genérico y el orientado a objetos.
- Portabilidad: Este lenguaje tiene un amplio rango de compiladores que pueden correr sobre diferentes tipos de plataformas computacionales.
- Soporte a tipo de datos estáticos y dinámicos: Permite el control de tipos de datos, ya sea en tiempo de compilación o en tiempo de ejecución.

2.7. Gecode

Gecode es un conjunto de herramientas para el desarrollo de sistemas y aplicaciones basados en restricciones. Gecode proporciona un solucionador de restricciones con el más moderno desempeño, mientras es modular y extensible. Este lenguaje de programación goza de las siguientes características [10]:

- Abierto: Gecode es totalmente abierto para programadores. Es fácil de acoplar con otros sistemas. Esto permite crear nuevas restricciones, nuevas estrategias de distribución y nuevos motores de búsqueda.
- Exhaustivo: Comprende un grupo de herramientas exhaustivas, como restricciones sobre enteros, booleanos, conjuntos y flotantes; modelamiento de capas por medio de C++; implementación de avanzadas heurísticas; entre otras.
- Eficiencia: Gecode ofrece un excelente rendimiento con respecto al tiempo de ejecución y el uso de memoria.
- Libre: Es un software libre con licencia del “Instituto Tecnológico de Massachusetts (MIT)”.
- Documentado: Contiene una documentación completa en su sitio web. [10]
- Paralelo: Tiene la capacidad de explotar múltiples núcleos.

Gecode presenta una librería de programación por restricciones implementada en C++ que integra técnicas novedosas para disminuir el tiempo que tarda en encontrar solución a problemas combinatorios. Por lo tanto, se convierte en una herramienta ideal para la implementación de éste trabajo de grado. Adicionalmente, Gecode permite un diseño modular el cual facilita la integración con otros lenguajes de programación existentes.

Capítulo 3

Estado de arte

Los trabajos académicos previamente realizados en el campo de planeación de estiba de contenedores, se centran en algoritmos de optimización acerca de la planificación de ranuras en los buques portacontenedores. Sin embargo, la inaccesibilidad a la información del problema de estibación, ha desafiado los diferentes enfoques de solución, hasta ahora propuestos. Estos trabajos pueden ser divididos en dos grupos principales: métodos basados en un único modelo del problema y métodos basados en la descomposición del problema en varios modelos de optimización. Entre estos dos enfoques, el segundo ha sido más exitoso, aunque la mayoría de la literatura se centra en métodos heurísticos, usando un único modelo.

3.1. Métodos basados en modelos únicos

Dentro de los métodos basados en un único modelo, Avriel et al (1998)[4] presentan un modelo 0-1 IP (Integer Programming) y un procedimiento de heurística suspensiva para la estiba de buques, la cual se restringe a una sola bahía del buque y resuelve el problema de estiba de contenedores de un solo tamaño, sin tomar en cuenta la estabilidad del buque. Esta heurística apunta a minimizar la sobre estiba en un problema multi-puerto, donde solo la carga de contenedores es permitida.

Dubrosky & Penn (2002)[9] presentaron un algoritmo genético bajo los mismos supuestos de Avriel et al. Sin embargo, su propuesta fue suficientemente flexible para incluir nuevas restricciones, y como resultado, instancias aleatorias de problemas con buques de 500 y 1000 TEUs, fueron resueltos, para 10 puertos, en 30 minutos.

Ambrosio & Sciomanchen (1998) [2] presentaron un modelo de programación por restricciones optimizado con una estructura de ramificación y poda (branch & bound). Aunque este modelo tuvo por objetivo estibar un buque completo, restricciones de estabilidad entre bahías pudieron ser obviadas, con el fin de asemejarse al problema de planeación de ranuras. Este método considera contenedores de 20 y

40 pulgadas, pero no tiene en cuenta los contenedores refrigerados, ni los tipo cúbicos. Su principal objetivo es minimizar la sobre estiba y el consumo, maximizando el número de contenedores cargados.

Otros trabajos sobre métodos basados en un único modelo, fueron propuestos por Aslidis (1984) [3], Botter & Brinati (1992)[5], Sciomachen & Tanfani (2003)[16] y Li et al (2008)[12], en donde se presentan modelos complejos que incluyen varias restricciones propuestas en el problema maestro de planeación. Aslidis introdujo heurísticas de apilamiento para minimizar sobre estiba, mientras Botter, Brinati y Li et al presentaron modelos 0-1 IP de programación entera. Adicionalmente, Botter y Brinati presentaron dos nuevas heurísticas para estibar contenedores debido a que su modelo IP no era escalable en instancias de la vida real. Sciomachen y Tanfani introdujeron un enfoque heurístico basado en problemas 3D de embalaje, en donde a diferencia de los demás métodos mencionados, se tuvo en cuenta los contenedores tipo cúbicos y no sólo los contenedores de 20 y 40 pulgadas.

3.2. Métodos basados en la descomposición del problema en varios modelos de optimización

Por otra parte, dentro de los métodos basados en la descomposición del problema en varios modelos de optimización, para la planeación de estiba de buques portacontenedores, Wilson & Roach (2000) [19], introdujeron la descomposición basada en bloques. Estos autores dividieron cada bahía de los buques en bloques, los que estaban distribuidos sobre y debajo de la cubierta y a menudo seguían el patrón de las tapas de escotilla. En resumen, Wilson y Roach propusieron un algoritmo tabú de búsqueda, para solucionar una versión de la planeación de estiba, que incluía ranuras de refrigeración, restricciones de tamaño de los contenedores, minimización de sobre estiba y restricciones de pilas con el mismo puerto de descarga. Sin ninguna evidencia de peso, aparte de resultados experimentales, los autores afirmaron encontrar soluciones óptimas cercanas y de rápida computación a instancias de problemas para buques de 688 TEUs. Kang & King (2002)[11] también adoptaron el concepto de descomposición en bloques y describieron un método de enumeración para resolver una versión simple de planeación de ranuras, donde sólo es tenido en cuenta la minimización de sobre estiba, la violación de estabilidad y el ordenamiento de contenedores de 40 pulgadas después que los pesos son considerados. Tal como Wilson y Roach, ninguna evaluación experimental independiente del algoritmo fue proporcionada.

Ambrosino et al (2006)[1] describen un modelo 0-1 IP para estibar óptimamente subconjuntos de bahías de contenedores, en buques con el mismo puerto de descarga. Este modelo minimiza el tiempo de estiba de contenedores de 20 y 40 pulgadas, y éstos son ordenados de acuerdo a su peso en cada pila. En la sección experimental, planes completos de estiba para buques de 198 a 2124 TEUs fueron generados, en donde los tamaños máximos de las bahías fueron de 20 y 120 TEUs, respectivamente. No

fue dado tiempo computacional para la solución de estos sub problemas. En trabajos posteriores, Ambrosino et al, presentan una heurística constructiva para solucionar los mismos sub problemas como uno único. Usando esta heurística, son capaces de estibar un buque de 5632 TEUs, en donde se usan 11.8 segundos en promedio para estibar todas las bahías, pero la disposición física del buque no es descrita en detalle.

Otros autores como Zhang et al (2005)[17] y Yoke et al (2009)[13] presentaron enfoques multi-fases donde los problemas resueltos durante la fase de planeación de ranuras, no son independientes unos de los otros.

Por último, Pacino et al(2011)[14] presentan un modelo de programación entera para la estiba de contenedores en áreas de estibación de barco, teniendo en cuenta múltiples puertos. Adicionalmente, Delgado et al (2012)[8] presentan un modelo de programación por restricciones para una óptima y veloz estiba de bahías en barcos contenedores. Este modelo se centra en el problema de estiba de contenedores ubicados por debajo de la cubierta.

3.3. Descripción del modelo en restricciones de Delgado et al (2012)[8].

Anteriormente, se mencionó que el modelo presentado por Delgado et al (2012) hace parte de una de las descomposiciones del problema total de estibación en buques portacontenedores. Por lo tanto, se explica brevemente cada una de las descomposiciones del problema de estiba. (Ver figura 1.1)

La primera fase del modelamiento general al problema de estiba de contenedores se denomina *Planeación Maestra* y tiene como responsabilidad distribuir los contenedores a cargar en secciones de bahías de la embarcación. Es decir, esta fase se encarga principalmente de separar conjuntos de contenedores y asignar esos conjuntos a determinadas secciones del barco, teniendo en cuenta que los contenedores de dichos conjuntos se cargan y se descargan en diferentes puertos [14].

La segunda fase del modelamiento general se denomina *Planeación de ranuras*. Esta fase tiene como responsabilidad asignar los contenedores a cargar en las sección de bahía, seleccionadas en la primera fase, en ranuras específicas. Es decir, esta fase procesa cada uno de los conjuntos de contenedores asignados a determinadas secciones de bahía y asigna cada contenedor del conjunto en una ranura de la bahía. Debido a las diferentes condiciones a las que se enfrentan los contenedores a estibar sobre y debajo de cubierta [8], el modelo utilizado para estibar contenedores por debajo de cubierta no puede ser el mismo para estibar por encima.

En este proyecto, la idea es mejorar lo propuesto por Delgado et al (2012)[8], ya que su trabajo sólo se centra en la estiba de contenedores por debajo de cubierta. Es decir, el proyecto a presentar en este documento se enfoca en la estiba de contenedores por encima de cubierta, ya que el modelo a extender solo estiba por debajo

Índices y constantes	Descripción
$Stack$	Conjunto de índices de pilas.
$Slots$	Conjunto de índices de bahías para los contenedores.
$Cont$	Conjunto de índices de contenedores.
$Slot^{\{A,F\}}$	Conjunto de espacios de popa y proa.
$Slot_k$	Conjunto de espacios por pila k.
$Slot_k^{\{A,F\}}$	Conjunto de espacios de popa y proa por pila k.
$Slot^{\{R,\neg R\}}$	Conjunto de espacios refrigerados y no refrigerados.
$Slot^{\neg RC}$	Conjunto de celdas no refrigeradas.
$Slot^{\{20,40\}}$	Conjunto de espacios para contenedores de 20 y 40 pies.
$Cont^{\{V,L\}}$	Contenedores virtuales (V) y cargados (L).
$Cont^{\{20,40\}}$	Índice del contenedor de 20 y 40 pies.
$Cont^{40\{A,F\}}$	Contenedor de 40 pies separados por proa y popa.
$Cont^{R\{20,40\}}$	Índice del contenedor refrigerados de 20 y 40 pies.
$Cont^{\neg R}$	Conjunto de contenedores no refrigerados.
$Weight_i$	Peso del contenedor i.
POD_i	Puerto de descarga del contenedor i.
$Length_i$	Largo del contenedor i.
$Height_i$	Altura del contenedor i.
$Cont^{P=p}$	Número de contenedores con puerto de descarga p.
$Cont^{\{W=w,H=h\}}$	Número de contenedores con peso w y alto h.
$Cont^{\{NC,HC\}}$	Número de contenedores normales (NC) y número de contenedores cubicos (HC).
$Stack_i^{\{W,h\}}$	Peso y altura limite de la pila i.
$Classes$	Conjunto de clases de pila.
$Class^i$	Conjunto de pilas de clase i.

Cuadro 3.1: Tabla de constantes.

de cubierta. Por lo tanto, se presentará una breve descripción del modelo general de restricciones presentado por Delgado et al (2012)[8].

Este modelo de restricciones cuenta con un conjunto de índices y de constantes utilizadas para la resolución del problema (Ver cuadro 3.1).

Para facilitar el modelamiento de restricciones y objetivos del problema, se adicionan unas variables auxiliares (Ver cuadro 3.2).

Adicionalmente, se tienen un conjunto de variables de decisión que representan un plan de estibación, para un conjunto de contenedores que pueden estibarse. Las variables se separan en dos subconjuntos. El primer subconjunto indica, por cada variable de decisión un contenedor que se puede estibar, donde el dominio de estas variables es definido por las ranuras o espacios. El segundo subconjunto indica, por cada variable de decisión una ranura, donde el dominio de estas variables es definido por los contenedores a estibar. Para utilizar el esquema de dos subconjuntos es ne-

Variables de decisión	Descripción
$C = \{C_1, \dots, C Cont \}$	$C_i \in Slots$, índice de la espacios en el $Cont_i$.
$S = \{S_1, \dots, S Slots \}$	$S_i \in Cont$, índice del contenedor en el $Slot_i$.
$L = \{L_1, \dots, L Slots \}$	$L_i \in Length$, largo del contenedor estibado en el $Slot_i$.
$H = \{H_1, \dots, H Slots \}$	$H_i \in Height$, alto del contenedor estibado en el $Slot_i$.
$W = \{W_1, \dots, W Slots \}$	$W_i \in Weight$, peso del contenedor estibado en el $Slot_i$.
$P = \{P_1, \dots, P Slots \}$	$P_i \in POD$, puerto de descarga del contenedor estibado en el $Slot_i$.
$HS = \{HS_1, \dots, HS Stack \}$	$HS_i \in Stack$, Limite de altura por pila.
$O^V \in \{0, \dots, Cont \}$	Número de contenedores sobre-estibados.
$O^U \in \{0, \dots, Stack \}$	Número de pilas usadas.
$O^P \in \{0, \dots, Stack POD \}$	Número de contenedores diferentes por cada pila.
$O^R \in \{0, \dots, Slot^R \}$	Número de contenedores no refrigerados estibados en espacios refrigerados.
$O \in \mathbb{N}$	Variable utilizada como costo total de estiba.
$C^V \subset C$	Contenedores virtuales.
$S_l^E \subset S$	espacios con las mismas características en la pila l.

Cuadro 3.2: Tabla de variables de decisión.

cesario que los contenedores sean iguales a las ranuras, por lo cual, se utilizan unos contenedores virtuales para llenar las ranuras faltantes.

Las restricciones desde (3.1) hasta (3.14) son reglas que se deben cumplir en el proceso de estiba.

$$channeling(C, S) \quad (3.1)$$

$$c_{fore(i)} = c_i + 1 \quad \forall i \in \{1, \dots, |Cont^{40A}|\} \quad (3.2)$$

$$element(S_i, L_i, Length) \quad \forall i \in Slots \quad (3.3)$$

$$element(S_i, H_i, Height) \quad \forall i \in Slots \quad (3.4)$$

$$element(S_i, W_i, Weight) \quad \forall i \in Slots \quad (3.5)$$

$$element(S_i, P_i, POD) \quad \forall i \in Slots \quad (3.6)$$

$$S_{pos(j)} = i \quad \forall i \in Cont^L \quad (3.7)$$

$$regular(Length_i^\pi R) \quad \forall \pi \in \{A, F\}, i \in Stacks \quad (3.8)$$

$$S_i \notin Cont^{20R} \quad \forall i \in Slots^R \quad (3.9)$$

$$S_i \notin Cont^{40R} \quad \forall i \in Slots^{RC} \quad (3.10)$$

$$S_i \in Cont^{20} \quad \forall i \in Slots^{20} \quad (3.11)$$

$$S_i \in Cont^{40} \quad \forall i \in Slots^{40} \quad (3.12)$$

$$\sum_{j \in Slots_i} h_j \leq h_{s_i} \quad \forall \pi \in \{A, F\}, i \in Stacks \quad (3.13)$$

$$\sum_{j \in Slots_i} w_j \leq Stack_i^j \quad \forall i \in Stacks \quad (3.14)$$

La restricción 3.1 canaliza dos subconjuntos de variables de decisión. La (3.2) garantiza que para un contenedor de 40 pies, la parte popa y proa se encuentren en la misma celda. Las restricciones (3.3), (3.4), (3.5) y (3.6) tienen como objetivo enlazar las variables de ranura, con las variables auxiliares de longitud, altura, peso y el puerto de descarga del contenedor estibado para una determinada ranura. La restricción (3.7) es para asignar contenedores cargados en su ranura predefinida. La (3.8) es una restricción regular para la popa y proa de cada pila. La restricción (3.9) y la (3.10) modelan la estiba de contenedores refrigerados. La (3.11) y (3.12) restringen los dominios de ranuras que sólo tienen capacidad para contenedores de 20 o 40 pies. La (3.13) limita la altura de cada pila en la ubicación delimitada. Por último, la restricción (3.14) restringe el peso de todos los contenedores estibados en una pila, para que estén dentro de los límites.

En el modelo de restricciones se calculan un conjunto de valores, con el fin de representar unos objetivos, éstos son:

1. Calcular la cantidad total de re-manipulaciones (movimientos realizados para estibar un contenedor en la carga o descarga del mismo).

$$O^v = \sum_{i \in Slots^A} OV(i) \quad (3.15)$$

2. Encontrar la sumatoria de ranuras utilizadas en una pila y calcular la suma de los resultados obtenidos por cada pila.

$$O^u = \sum_{k \in Stacks} \left(\sum_{j \in Slots_k} P_j > 0 \right) \quad (3.16)$$

3. Calcular el número de contenedores estibados que se descargan en diferentes puertos en cada pila.

$$p = \sum_{k \in Stacks} \left(\sum_{\rho \in POD} \sum_{j \in Slots_k} P_j = \rho \right) > 0 \quad (3.17)$$

4. Contar el número de contenedores no refrigerados en ranuras refrigeradas.

$$O^r = \sum_{i \in Slots^R} (S_i \in Cont^R) \quad (3.18)$$

En el modelo se pretende minimizar una función de costo. La función de costo es la suma de los valores que representan los objetivos multiplicados con un respectivo peso de costo.

$$O = 100 * O^v + 20 * O^p + 10 * O^u + 5 * O^r \quad (3.19)$$

Capítulo 4

Descripción del problema

El problema que se intenta resolver con este proyecto, tiene como objetivo final obtener un plan de estiba que ayude a los coordinadores de estiba a realizar mejor su trabajo. Este proyecto empieza equivalente al de Delgado et al (2012)[8], en la segunda fase del modelo general, *planeación de ranuras* (Ver figura 1.1) que hace parte de una de las descomposición heurísticas que se realizan para poder tratar mejor el problema de estibar contenedores (Ver sección 3.3). El enfoque en este proyecto se centra en asignar contenedores en las locaciones que se encuentran sobre la cubierta del buque y denominamos éste problema como (**CSPADL** - Container Stowage Problem for Above Deck-Locations).

4.1. Reglas y límites del problema de estiba

Un buque portacontenedores contiene cerca de 100 locaciones. Lo anterior, implica que la fase de *planeación de ranuras* tiene que resolver 100 instancias independientes. Por otro lado, sólo se tiene 10 minutos para generar completamente un plan de estiba. Es decir, en 10 minutos se deben ejecutar, tanto la primera fase *Planeación Maestra*, como la segunda fase *Planeación de Ranuras*, de la descomposición heurística. La idea con este proyecto, es que se puedan resolver las instancias de la segunda fase en menos de 1 segundo.

En este proyecto se hace una abstracción de las restricciones y los objetivos reales del problema de estibación de contenedores. Este trabajo se limita a utilizar sólo contenedores de 20 y 40 pies de largo, de 8.6 y 9.6 pies de alto, refrigerados y no refrigerados. Además, en el proceso de estiba de contenedores es obligatorio cumplir con determinadas reglas de estibación, éstas son:

1. Las celdas asignadas deben formar pilas. Los contenedores deben estar soportados sobre otro contenedor o por el piso del buque, no deben estar en el aire.

2. Un contenedor de 20 pies no debe apilarse sobre uno de 40 pies.
3. Un contenedor de 20 pies refrigerado debe ser asignado en una ranura o espacio refrigerado. Un contenedor refrigerado de 40 pies debe ser asignado a una celda que posea por lo menos una ranura refrigerada.
4. Algunas celdas del barco sólo permiten estibar contenedores de 20 pies y pueden existir celdas donde solamente se pueden estibar contenedores de 40 pies.
5. La suma de las alturas y los pesos de cada contenedor estibado en una pila, debe cumplir con los límites de peso y altura de la pila.
6. Los contenedores ya estibados en el buque no pueden ser estibados en otras ranuras.
7. No pueden existir contenedores sobre-estibados.
8. Los contenedores deben estar ordenados en la pila de abajo hacia arriba de mayor a menor peso.

Adicionalmente, es necesario que el plan de estiba evite costos innecesarios. Por lo cual, se plantean unos objetivos a minimizar, éstos son:

1. El número de contenedores que no se estiban en el barco.
2. El número de puertos de descarga de los contenedores estibados en cada pila.
3. El número de pilas usadas.
4. El número de contenedores no refrigerados estibados en ranuras refrigeradas.
5. Distancia entre el centro de gravedad vertical real y el centro de gravedad vertical ideal de cada pila.

4.2. Proceso de estiba y trincaje en la computación

En la computación, **CSPADL** es un problema NP-Completo. Lo anterior, se basa en que se puede reducir del problema **Bin-Packing**, que es NP-completo, a **CSPADL**. Para realizar la reducción es necesario ver los "Bins" como contenedores, en donde el alto y el ancho corresponden a los límites de altura y peso de una pila en el buque; no se tiene en cuenta que existan ranuras refrigeradas. Por otro lado, todos los elementos a organizar en los "Bins", podrían verse como elementos con una altura y un ancho correspondientes al alto y al peso de un contenedor. Estos elementos estarían definidos como contenedores no refrigerados, de 40' pies y con el mismo puerto de descarga.

Adicionalmente, se debe tener en cuenta que las características de peso y de altura de los elementos deben concordar con las características de peso y altura del “Bin”. Por lo anterior, encontrar una solución óptima para el problema de **Bin-Packing** tendrá como resultado encontrar una solución óptima para **CSPADL**.

Capítulo 5

Introducción a Modelos

En esta sección se pretende dar una introducción al modelo creado en esta investigación. En el modelo creado, para gestionar con más facilidad los contenedores de 40 pies, éstos se han dividido en 2 contenedores de 20 pies, que irán ubicados en la proa y la popa de la misma celda. Adicionalmente, las bahías en el buque son indexadas de popa a proa y de abajo hacia arriba en la pila y continúan en las demás pilas de izquierda a derecha en una locación del barco (Figura 5.1 y 5.2).

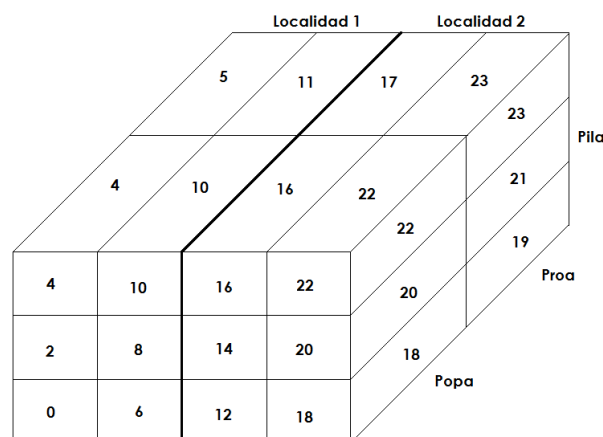


Figura 5.1: Índices de espacios (Slots)

Para obtener resultados factibles al problema de estibar contenedores se utiliza en el modelo operaciones lógicas conocidas como restricciones, que permiten imponer reglas al proceso de estiba. Este modelamiento por restricciones requiere básicamente de 3 pasos, los cuales son, la definición de las variables de decisión y sus respectivos dominios, restricciones pertinentes y la estrategia de distribución.

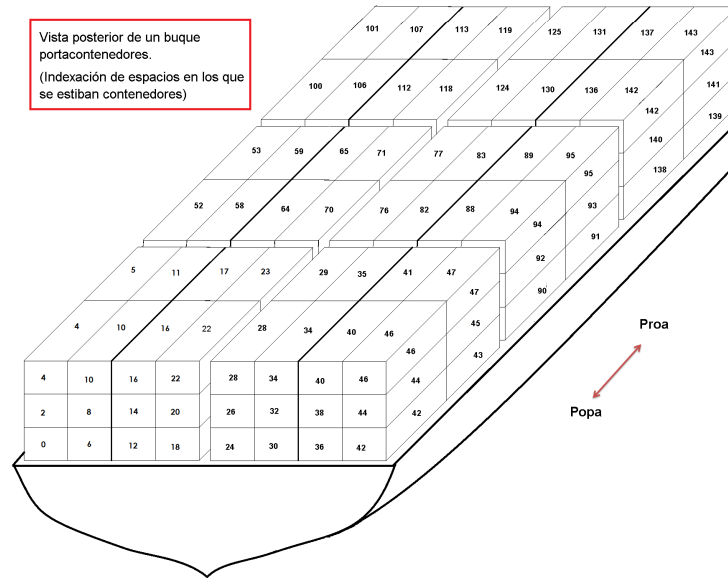


Figura 5.2: Indexación de espacios en el buque

5.1. Abstracciones de restricciones

A continuación, se muestran unas abstracciones de restricciones que serán utilizadas en el modelo creado en éste documento.

Restricciones elementales

Estas restricciones tienen la siguiente forma:

$$element(X, Y, C) = \{(e, f) | e \in D(X), f \in D(Y), f = C_e\}$$

Donde X es una variable entera, Y es una variable con dominio finito y $C = \{C_1, \dots, C_n\}$ un conjunto de constantes. La restricción implica que Y es igual a la X -ésima constante de C .

Restricciones regulares

Estas facilitan el uso de expresiones regulares:

$$regular(X, M) = \{(d_1, \dots, d_n) | \forall i. d_i \in D(X_i), d_1 \dots d_n \in L(M)\}$$

Donde X es un conjunto de variables con dominio $D(X_i)$ tal que $1 < i < n$ y M es una expresión regular que reconoce $L(M)$.

Restricciones Max

Este tipo de restricción tiene la siguiente forma:

$$\begin{aligned} \max(H, I, J) = \{ (e, f, g) \mid \\ e \in D(H), f \in D(I), g \in D(J), \\ e > f \Rightarrow g = e, f \geq e \Rightarrow g = f \} \end{aligned}$$

Donde la restricción implica que la variable J será el valor más grande entre H e I .

Restricciones Channel

Este tipo de restricción se puede expresar de dos maneras:

- La primera abstracción es cuando H es una variable no escalar e I es una variable escalar. Esta abstracción tiene la siguiente forma:

$$\text{channel}(H, I) = \{ (e, f) \mid e \in D(H), f \in D(I), e = f \}$$

Esta restricción se utiliza para realizar un cambio de una variable escalar a una variable no escalar.

- La segunda abstracción es cuando X y Y son conjuntos de variables. Esta abstracción se presenta formalmente de la siguiente forma:

$$\begin{aligned} \text{channel}(X, Y) = \{ (e_1, \dots, e_n, f_1, \dots, f_n) \mid \\ \forall i, j. e_i \in D(X_i), f_i \in D(Y_j), e_i = j \Leftrightarrow f_j = i \} \end{aligned}$$

Esta restricción enlaza consistentemente dos conjuntos de variables del mismo tamaño.

Restricciones Expr

Este tipo de restricción tiene la siguiente forma:

$$R = \text{expr}(\text{Expres}) = \{ h \mid h \in D(R) \}$$

Donde la restricción ejecuta la expresión Expres y el resultado lo guarda en R .

Restricciones Ite

Este tipo de restricción se expresa formalmente así:

$$Ite(H, I, J, K) = \{(b, x, y, z) \mid \\ b \in D(H), i \in D(I), j \in D(J), z \in D(K), z = b?x : y\}$$

En esta restricción se asigna $z = x$ si b es verdadero y se asigna $z = y$ si b es falso.

Restricciones Count

Este tipo de restricción tiene la siguiente forma:

$$Count(X, Y, R, Z) = \#(i \in \{0, \dots, |X| - 1\} | X_i \in Y) \sim_R Z$$

Donde esta restricción cuenta la cantidad de elementos X que concuerden con los elementos de Y y ejecuta una relación R , que puede ser $<, >, =, \neq$, entre otras, con respecto a la variable Z .

Restricciones atmost

Este tipo de restricción tiene la siguiente forma:

$$atmost(X, Y, Z) = \#(i \in \{0, \dots, |X| - 1\} | X_i = Y) \leq Z$$

Ésta restricción cuenta la cantidad de elementos X que concuerden con el valor Y y ejecuta una relación \leq con respecto al valor Z .

Restricciones exactly

Este tipo de restricción tiene la siguiente forma:

$$exactly(X, Y, Z) = \#(i \in \{0, \dots, |X| - 1\} | X_i = Y) = Z$$

Donde ésta restricción cuenta la cantidad de elementos X que concuerden con el valor Y y ejecuta una relación $=$ con respecto al valor Z .

Restricciones lex_{less}

Este tipo de restricción tiene la siguiente forma:

$$lex_{less}(X, Y)$$

Donde X y Y son conjuntos de n componentes $\langle X_0, \dots, X_{n-1} \rangle$ y $\langle Y_0, \dots, Y_{n-1} \rangle$. X es lexicográficamente menor a Y cuando $X_0 < Y_0$ o $X_0 = Y_0 \wedge \langle X_1, \dots, X_{n-1} \rangle$ es lexicográficamente menor a $\langle Y_1, \dots, Y_{n-1} \rangle$.

5.2. Conjunto de constantes

Para facilitar la construcción de las restricciones y de los objetivos a alcanzar, expuestos en la sección 4.1, se plantean un conjunto de índices y constantes que sirven de apoyo al modelo contruido en éste documento. Todos los conjuntos de índices que se presentarán son un subconjunto de enteros a excepción de la constante $GCSY$ que puede ser un dato no escalar. Cada conjunto representa características, ya sean de secciones del barco o de los contenedores (Ver cuadro 5.1).

Sobre éste modelo se busca encontrar el centro de gravedad por cada una de las pilas estibadas en el barco y se debe obtener la diferencia entre el centro de gravedad encontrado, con respecto al centro de gravedad ideal de cada pila. El centro de gravedad ideal de una pila en el eje Y se encuentra dado por la formula $(Stack_i^H/3)/(Stack_i^H/maxLevel(i))$ donde $maxLevel(i)$ es el nivel máximo de la pila i . En el cálculo del centro de gravedad sólo se busca la coordenada Y , debido a que la coordenada sobre el eje X no brinda información relevante para la estiba y trincaje de contenedores.

Para calcular el centro de gravedad de un conjunto de contenedores agrupados en una pila, es necesario proponer un sistema de coordenadas sobre cada pila. El sistema de coordenadas para cada pila, empezará con el punto $(0, 0)$ sobre el centro del contenedor ubicado en la parte inferior de la pila en la sección de la popa. La coordenada aumentará en el eje X de la popa a la proa en la pila y en el eje Y de la parte inferior a la parte superior de la pila. (Ver figura 5.3)

Anteriormente, se mencionó que los contenedores de 40 pies estarían representados por medio de dos contenedores de 20 pies. Para evitar problemas al calcular datos que requieran el peso de los contenedores de 40 pies que ya estén estibados, la constante $Weight_i$ de contenedores de 40 pies deberá ser dividida entre los dos contenedores de 20 pies. El contenedor de 20 pies que representa la división de la popa tendrá un peso igual a $Ceil(Weight_i/2)$ y el contenedor de 20 pies que representa la proa será igual a $Weight_i - Ceil(Weight_i/2)$

Los capítulos 6 y 7 tienen como objetivo explicar a detalle el modelo realizado y una extensión del modelo investigado. Para el modelo y su extensión, se explicarán las variables de decisión utilizadas y sus respectivos dominios, además, se explicarán

Índices y constantes	Descripción
$Slots$	Conjunto de índices de espacios para los contenedores.
$Cont$	Conjunto de índices de contenedores.
$Stack$	Conjunto de índices de pilas.
POD	Conjunto de puertos de descarga.
$Slot_k$	Conjunto de espacios por pila k .
$Slot_k^{A,F}$	Conjunto de espacios de popa y proa por pila k .
$Slot_R$	Conjunto de espacios refrigerados.
$Slot^{NR}$	Conjunto de espacios no refrigerados.
$Slot^{NRC}$	Conjunto de celdas no refrigeradas.
$Cont^L$	Conjunto de contenedores cargados.
$Cont_K$	Conjunto de contenedores en la pila k .
$Cont^{\{20,40\}}$	Índice del contenedor de 20 y 40 pies.
$Cont_{A,F}^{40}$	Índice del contenedor de 40 pies separados por proa y popa.
$Cont_R^{\{20,40\}}$	Índice del contenedor refrigerados de 20 y 40 pies.
$Length_i$	Largo del contenedor i .
$Weight_i$	Peso del contenedor i .
$Height_i$	Altura del contenedor i .
POD_i	Puerto de descarga del contenedor i .
$Stack_i^W$	Peso límite de la pila i .
$Stack_i^H$	Altura límite de la pila i .
$Cont^V$	Índice del contenedor virtual.
$Cont^{Load}$	Conjunto de contenedores a cargar.
$GCSY$	Centro de gravedad ideal en el eje Y .
$ContNonReefer$	Conjunto de variables booleanas del tamaño $ Cont $, donde cada elemento representa la relación $ContNonReefer_i = Cont_i \in Cont_N R$
$ContEW_i$	Número de contenedores con igual peso i .
$ContEP_i$	Número de contenedores con igual puerto de descarga i .
$ContEH_i$	Número de contenedores con igual altura i .
$ContEL_i$	Número de contenedores con igual largo i .
$CountStacks$	Número total de pilas del buque.

Cuadro 5.1: Tabla de constantes.

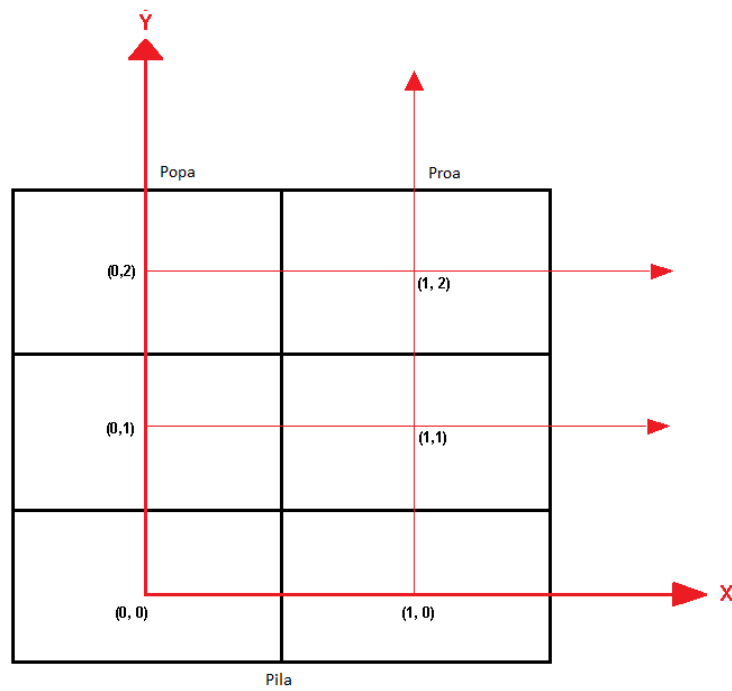


Figura 5.3: Coordenadas para cálculo de centro de gravedad

las restricciones que se aplican sobre estas variables de decisión y las estrategias de distribución utilizadas. El capítulo 9 mostrará los resultados obtenidos por el modelo propuesto y por la extensión del mismo modelo. Estos resultados son evaluados y comparados, y adicionalmente, cabe resaltar que fueron obtenidos a partir de instancias reales utilizadas en la estiba de contenedores en los puertos costeros.

Capítulo 6

Modelo original

En este modelo se propone la idea de utilizar un contenedor virtual que tiene como objetivo representar un contenedor que no se cargará en el espacio que lo contenga. Las características como el largo, ancho, alto y puerto de descarga serán iguales a 0. Además, el índice del contenedor virtual es igual a 0 ($Cont^V = 0$).

6.1. Variables de decisión

Sobre la variables de decisión actuarán las restricciones que se propondrán más adelante. Adicionalmente, estas variables modelarán los posibles resultados obtenidos al final de proceso (Ver cuadro 6.1).

Las variables de decisión utilizadas para calcular el costo de las soluciones encontradas, son utilizadas sobre el proceso de propagación. (Ver cuadro 6.2)

6.2. Restricciones del modelo

Para el modelo se plantean un conjunto de restricciones que tienen como objetivo disminuir el espacio de búsqueda de posibles soluciones. En esta sección se presentan y se detallan el conjunto de restricciones modeladas.

$$atmost(S, i, 1) \quad \forall i \in Cont | i \neq 0 \quad (6.1)$$

$$element(S_i, L_i, Length) \quad \forall i \in Slots \quad (6.2)$$

$$element(S_i, H_i, Height) \quad \forall i \in Slots \quad (6.3)$$

Variables de decisión	Descripción
$S = \{S_1, \dots, S Slots \}$	$S_i \in \text{Cont}$, índice del contenedor en el $Slot_i$.
$L = \{L_1, \dots, L Slots \}$	$L_i \in \text{Length}$, largo del contenedor estibado en el $Slot_i$.
$H = \{H_1, \dots, H Slots \}$	$H_i \in \text{Height}$, alto del contenedor estibado en el $Slot_i$.
$W = \{W_1, \dots, W Slots \}$	$W_i \in \text{Weight}$, peso del contenedor estibado en el $Slot_i$.
$WD = \{WD_1, \dots, WD Slots \}$	$WD_i \in \text{Weight}$, peso del contenedor estibado en el $Slot_i$. (Arreglo de variables no escalares).
$P = \{P_1, \dots, P Slots \}$	$P_i \in \text{POD}$, puerto de descarga del contenedor estibado en el $Slot_i$.
$NRSR$ $\{NRSR_1, \dots, NRSR Slots^R \}$	$NRSR_i \in \{0, 1\}$, Indica si se carga un contenedor no refrigerado en un ranura refrigerada.
$HS = \{HS_1, \dots, HS Stack \}$	$HS_i \in \text{Stack}$, límite de altura por pila.
$CFEUA$ $\{CFEUA_1, \dots, CFEUA Slots/2 \}$	$CFEUA_i \in \{0, 1\}$, contenedor de 40 pies estibado en la popa.
$CFEUF$ $\{CFEUF_1, \dots, CFEUF Slots/2 \}$	$CFEUF_i \in \{0, 1\}$, contenedor de 40 pies estibado en la proa.
$OPT = \{OPT_1, \dots, OPT Stack \}$	$OPT_i \in \text{Cont}_K$, número de contenedores diferentes en la pila i.
$NVC = \{NVC_1, \dots, NVC Slots \}$	$NVC_i \in \{0, 1\}$, Ranuras o espacios asignados a contenedores no virtuales.
$WT = \{WT_1, \dots, WT Stack \}$	$WT_i \in \{0, \text{Stack}_i^W\}$, peso total de los contenedores estibados en la pila i.
$GCenY$ $\{GCenY_{1,1}, \dots, GCenY Stack Slots \}$	$GCenY_{i,s} \in \{0, \text{maxLevel}(i)\}$, centro de gravedad en eje Y para la pila i y el slot s.
$GCY = \{GCY_1, \dots, GCY Stack \}$	$GCY_i \in \{0, \text{maxLevel}(i)\}$, centro de gravedad en el eje Y.
$GCD = \{GCD_1, \dots, GCD Stack \}$	$GCD_i \in \{0, \text{maxLevel}(i)\}$, distancia al centro de gravedad ideal por pila.
$OUT = \{OUT_1, \dots, OUT Stack \}$	$OUT_i \in \text{Cont}^{\{20,40\}}$, número de contenedores i de 20 y 40 pies en la pila k.
$OUTE$	$0 \leq OUTE \leq Stack $, número de pilas vacías.
CS	$0 \leq CS \leq Cont $, número de contenedores estibados.
$C40F$	$0 \leq C40F \leq CFEUF $, número de contenedores de 40 pies estibados en la proa.

Cuadro 6.1: Tabla de variables de decisión.

Variables de decisión	Descripción
$OCNS$	Variable con el número de contenedores no estibados en el barco (Objetivo 1 de la sección 4.1).
OP	Variable con la sumatoria del número de contenedores diferentes por cada pila (Objetivo 2 de la sección 4.1).
OU	Variable con el número de pilas usadas (Objetivo 3 de la sección 4.1).
OR	Variable con el número de contenedores no refrigerados, estibados en ranuras refrigeradas (Objetivo 4 de la sección 4.1).
$OGCTD$	Variable con la sumatoria de los pesos otorgados por la distancia al centro de gravedad ideal por cada pila (Objetivo 5 de la sección 4.1).
O_{old}	Es un arreglo que contiene a las variables $OCNS$, OP , OU , OR y $OGCTD$ las cuales están ordenadas por prioridad. Donde $OCNS$ es el objetivo mas prioritario y $OGCTD$ es el menos prioritario. Este arreglo de variables contiene la mejor solución encontrada en búsquedas anteriores.
O_{new}	Es un arreglo muy parecido a O_{old} , se diferencia en que es la mejor solución encontrada en la búsqueda actual.

Cuadro 6.2: Tabla de variables de decisión utilizadas para calcular el costo de la solución.

$$element(S_i, W_i, Weight) \quad \forall i \in Slots \quad (6.4)$$

$$element(S_i, P_i, POD) \quad \forall i \in Slots \quad (6.5)$$

$$element(S_{Slots_i^R}, NRSR_i, ContNonReefer) \quad \forall i \in Slot^R \quad (6.6)$$

$$channel(WD_i, W_i) \quad \forall i \in Slots \quad (6.7)$$

$$S_{pos(i)} = i \quad \forall i \in Cont^L \quad (6.8)$$

$$S_i \notin Cont_A^{40} \quad \forall i \in Slot_k^F \wedge \forall k \in Stacks \quad (6.9)$$

$$S_i \notin Cont_F^{40} \quad \forall i \in Slot_k^A \wedge \forall k \in Stacks \quad (6.10)$$

$$CFEUA_{getCell(i)} = expr(L_i == 40) \quad \forall i \in Slot_k^A \quad (6.11)$$

$$CFEUA_{getCell(i)} = 1 \Rightarrow S_{i+1} = S_i + 1 \quad \forall i \in Slot_k^A \quad (6.12)$$

$$CFEUF_{getCell(i)} = expr(L_i == 40) \quad \forall i \in Slot_k^F \quad (6.13)$$

$$CFEUF_{getCell(i)} = 1 \Rightarrow S_{i-1} = S_i - 1 \quad \forall i \in Slot_k^F \quad (6.14)$$

$$P_i \geq P_{i+2} \geq \dots \geq P_{i+n} \quad \forall k \in Stack | i = Min(Slot_k^A), n = Max(Slot_k^A) \quad (6.15)$$

$$P_i \geq P_{i+2} \geq \dots \geq P_{i+n} \quad \forall k \in Stack | i = Min(Slot_k^F), n = Max(Slot_k^F) \quad (6.16)$$

$$WT_i = \sum_{j \in Slots_i} WD_j \quad \forall i \in Stacks \quad (6.17)$$

$$GCenY_{i,s} = WD_s * getPosY(s) \quad \forall i \in Stacks \quad \wedge \quad s \in Slots_i \quad (6.18)$$

$$GCY_i = (\sum GCenY_{i,s}) / (WT_i + 0,001) \quad \forall i \in Stacks \quad \wedge \quad s \in Slots_i \quad (6.19)$$

$$PenY_i = GCY_i - GCSY \quad \forall i \in Stacks \quad (6.20)$$

$$Max(PenY_i, 0, GCD_i) \quad \forall i \in Stacks \quad (6.21)$$

$$regular(Length_i^\alpha, R) \quad \forall \alpha \in \{A, F\} \wedge i \in Stack \quad (6.22)$$

$$\sum_{j \in Slots_i^\alpha} H_j \leq HS_i \quad \forall \alpha \in \{A, F\} \wedge i \in Stacks \quad (6.23)$$

$$\sum_{j \in Slots_i} W_j \leq Stack_i^W \quad \forall i \in Stacks \quad (6.24)$$

$$W_i + W_{i+1} \geq W_{i+2} + W_{i+3} \geq \dots \geq W_{i+n} + W_{i+n+1} \quad (6.25)$$

$$\forall k \in Stack | i = Min(Slot_k^A), n = Max(Slot_k^A)$$

$$OPT_k = distinctPOD(Slot_k) \quad \forall k \in Stacks \quad (6.26)$$

$$S_i \notin Cont_R^{20} \quad \forall i \in Slot^{NR} \quad (6.27)$$

$$S_i \notin Cont_R^{40} \quad \forall i \in Slot^{NRC} \quad (6.28)$$

$$S_i \in Cont^{20} \quad \forall i \in Slot^{20} \quad (6.29)$$

$$S_i \in Cont^{40} \quad \forall i \in Slot^{40} \quad (6.30)$$

$$NVC_i = expr(S_i! = 0) \quad \forall i \in Slots \quad (6.31)$$

$$CS = \sum_{i \in Slots} NVC_i \quad (6.32)$$

$$C40F = \sum_{i \in Slot_k^F} CFEUF_{getCell(i)} \quad (6.33)$$

$$Count(L_i, \{20, 40\}, =, OUT_k) \quad \forall i \in Slot_K^{40} \quad (6.34)$$

$$Count(OUT_k, \{0\}, =, OUTE) \quad \forall k \in Stack \quad (6.35)$$

$$atmost(W, i, ContEW_i) \quad \forall i \in ContEW | i \neq 0 \quad (6.36)$$

$$atmost(P, i, ContEP_i) \quad \forall i \in ContEP | i \neq 0 \quad (6.37)$$

$$atmost(H, i, ContEH_i) \quad \forall i \in ContEH | i \neq 0 \quad (6.38)$$

$$atmost(L, i, ContEL_i) \quad \forall i \in ContEL | i \neq 0 \quad (6.39)$$

La restricción (6.1) implica que un contenedor no puede estar en diferentes ranuras a excepción del contenedor virtual. Las restricciones <elementales> que se encuentran referenciadas del (6.2) al (6.5) son para generar una asociación entre las ranuras y las características de longitud, altura, peso y puerto de descarga de los contenedores que se estibarán en dichas ranuras. La restricción (6.6) que también es <elemental> se utiliza para identificar cuando un contenedor no refrigerado se estiba en un espacio refrigerado. Para poder realizar el cambio de una variable escalar, donde se guarda el peso del contenedor, a una variable no escalar se utiliza la restricción (6.7). Se presenta la restricción (6.8) para cargar los contenedores abordó del barco, esta restricción hace cumplir la regla 6. Las restricciones (6.9) y (6.10) son para que los contenedores de 20 pies que representan la proa y la popa de un contenedor de 40 pies se estiben respectivamente en las ranuras de proa y popa de una misma celda. Se modelan las restricciones del (6.11) al (6.14) que tienen como función que los dos contenedores de 20 pies que representan a uno 40 pies, queden estibados en la proa y popa de la misma celda, donde $getCell(i)$ tiene como tarea obtener la celda de la ranura i . Las restricciones (6.15) y (6.16) tienen como objetivo evitar contenedores sobre-estibados, un contenedor se considera sobre-estibado en una pila, cuando éste contiene un puerto de descarga mayor a cualquiera de los puertos de descarga de los

contenedores que se encuentren por debajo del mismo. Estas restricciones cumplen la regla 7. La restricción (6.17) se utiliza para obtener el peso total de los contenedores estibados por pila. Para calcular el centro de gravedad de las ranuras de una pila en el eje Y se utiliza la restricción (6.18); no se calcula el centro de gravedad en eje x debido a que no brinda información relevante. El calculo del centro de gravedad de la pila en el eje Y se realiza por medio de la restricción (6.19). La restricción (6.20) se utiliza para calcular la penalización que corresponde a la distancia entre el centro de gravedad calculado para la pila en el eje Y y el centro de gravedad ideal de la pila el cual es representado por medio de $GCSY$. La distancia del centro de gravedad para cada pila se obtiene del máximo valor entre la penalización por pila y la constante 0, esta distancia se obtiene por medio de la restricción (6.21). La restricción (6.22) plantea que las variables de longitud que pertenecen a una misma pila deben cumplir con la expresión regular $R = 20*40*0^*$, esta restricción hace cumplir las reglas de estibación 1 y 2. La restricción (6.23) especifica el límite de altura para cada pila. El límite de peso para cada pila es especificado por medio de la restricción (6.24). Las restricciones (6.23) y (6.24) hacen cumplir la regla 5 de estibación. Se modela la restricción (6.25) la cual implica que los contenedores de una pila deben estar ordenados de abajo hacia arriba y de mayor a menor peso, donde $Min(x)$ y $Max(x)$ especifica el mínimo y máximo índice de conjuntos de ranuras que pertenecen a una pila y que aún no se han cargado, esta restricción cumple la regla 8. La restricción (6.26) tiene como objetivo contar por cada pila, la cantidad de contenedores estibados con diferentes puertos de descarga. Las restricciones (6.27) y (6.28) implican que no es posible estibar contenedores refrigerados en ranuras no refrigerados, estas restricciones hacen cumplir la regla 3. Las restricciones (6.29) y (6.30) restringen el dominio de las ranuras de 20 y 40 pies, indicando que sólo se pueden cargar contenedores de 20 y 40 pies respectivamente, estas restricciones hacen cumplir la regla 4. La restricción (6.31) es para representar los espacios o ranuras que no han sido estibados con el contenedor virtual. La restricción (6.32) cuenta la cantidad de contenedores no virtuales estibados. La restricción (6.33) cuenta la cantidad de contenedores que hacen parte de la proa de los contenedores de 40 pies y que han sido estibados. La restricción (6.34) cuenta la cantidad de contenedores de 20 y 40 pies estibados por pila. La restricción (6.35) se encarga de obtener la cantidad de pilas vacías. Las restricciones (6.36), (6.37), (6.38) y (6.39) tienen como objetivo limitar el número de contenedores a estibar de acuerdo a características como el peso, el puerto de descarga, la altura y el largo de los contenedores.

$$OCNS = |Cont^L| + |Cont^{Load}| - (CS - C40F) \quad (6.40)$$

$$OU = CountStacks - OUTE - getMinUseStack() \quad (6.41)$$

$$OP = \sum_{i \in Stacks} OPT_i \quad (6.42)$$

$$OR = \sum_{i \in Slots^R} NRSR_i - getMinSlotsR() \quad (6.43)$$

$$OGCTD = \left(\sum_{i \in Stacks} GCD_i \right) * 100 \quad (6.44)$$

$$lex_{less}(O_{old}, O_{new}) \quad (6.45)$$

Las restricciones utilizadas para calcular el costo de las soluciones, se encuentran referenciadas del (6.40) al (6.45). A continuación, se detalla cada una de las restricciones de costo.

La restricción (6.40) cuenta el costo de contenedores no estibados en el buque. El costo de contenedores no estibados se obtiene del total de contenedores menos la diferencia entre la cantidad de contenedores no virtuales estibados y los contenedores que hacen parte de la proa de los contenedores de 40 pies. Para obtener la cantidad de pilas utilizadas, se usa la restricción (6.41) que obtiene la cantidad de pilas menos la cantidad de pilas no usadas y adicionalmente resta *getMinUseStack()* que es el mínimo número de pilas a utilizar dependiendo de la cantidad de contenedores estibados obtenidos de la variable *CS*. La restricción (6.42) suma la cantidad de contenedores con puerto de descarga diferente en una misma pila. La restricción (6.43) suma la cantidad de contenedores no refrigerados estibados en espacios refrigerados y resta *getMinSlotsR()* que es el mínimo número de ranuras refrigeradas posible en las cuales se pueden estibar contenedores no refrigerados. El valor de *getMinSlotsR()* se obtiene con base en el mínimo número de pilas a utilizar *getMinUseStack()*. La restricción (6.44) se encarga de obtener el peso del centro de gravedad, este calculo consiste en la suma de los centros de gravedad de cada pila multiplicado por la constante 100. Por último, la restricción (6.45) verifica que se encuentren mejores soluciones teniendo en cuenta la prioridad de los objetivos.

6.3. Estrategia de distribución

Para este modelo se utilizan tres estrategias de distribución. En todas las estrategias se hacen uso de las variables auxiliares *P*, *L*, *W* y *H* y por último se usa la variable *S*. Adicionalmente, sobre la variable *S*, en las estrategias de distribución, se aplica un rompimiento simétrico de valores, que se da al intentar estibar en una ranura, contenedores que tienen las mismas características.

Para un mejor rendimiento, al momento de realizar la distribución de las variables auxiliares P, L, W y H se tiene en cuenta la cantidad de valores diferentes que hay en el dominio de dichas variables. De esta forma, se distribuye primero sobre las variables auxiliares que contengan mas valores diferentes en su dominio. Sí la cantidad de valores a distribuir sobre las variables son dos, el valor nulo u otro valor, la variable auxiliar no se distribuye. Es decir, sí sólo existen contenedores de 20 pies en la instancia a probar, el dominio de las variables auxiliares L_i será $\{0, 20\}$ y para este caso en particular no se distribuirá sobre las variables L_i .

6.3.1. Estrategia de distribución por pilas específicas

La estrategia de distribución por pilas específicas consiste en agrupar conjuntos de valores, en donde cada conjunto representa una pila y los valores representan los índice de unos espacios en el buque portacontenedores. Los índices son utilizados para poder asignar el puerto de descarga, el largo, el peso, la altura y el contenedor en un espacio o ranura del buque. (Ver figura 6.1)

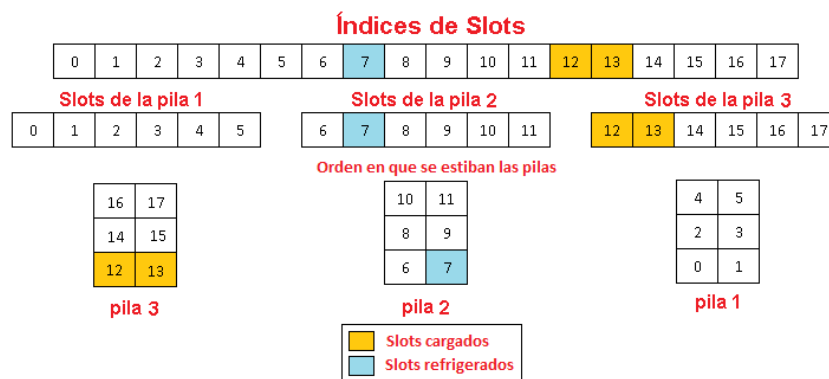


Figura 6.1: Índices por pila específica

La estrategia de distribución de agrupar por pilas permite fácilmente asignar valores de contenedores a una pila y la estrategia no continúa con la siguiente pila hasta que la primera pila, en la que se están asignando contenedores, cumpla con todas las restricciones del modelo. Para intentar utilizar la menor cantidad de pilas, se realiza la distribución, primero, sobre los pilas a las cuales ya se le han cargado contenedores y posteriormente, se buscan las pilas con menor cantidad de espacios o ranuras refrigeradas.

Sobre cada pila, la estrategia de distribución escoge los índices i del espacio o ranura a estibar, esta selección toma primero los índices del nivel inferior al nivel superior de la pila y en cada nivel de la pila avanza de popa a proa. Como se menciono anteriormente, se hace uso de las variables auxiliares P, L, W y H para disminuir el espacio de búsqueda, de esta forma se escoge el arreglo de variables cuyas variables tengan más valores en su dominio.

En la distribución, las variables auxiliares P_i de la pila, se asignan al principio con el máximo valor de P para el espacio i , intentando cumplir las restricciones (6.15) y (6.16). Las variables auxiliares L_i , intentado cumplir con la restricción (6.22), escogen primero los de largo 20, luego los de 40 y por último los de largo 0. Las variables auxiliares W_i , intentando cumplir con la restricción (6.25), escogen al principio el máximo valor de W para el espacio i . Igualmente, Las variables H_i , asignan el máximo valor de H para el espacio i . Por último, se intenta asignar la variable S buscando así, obtener resultado factibles para la pila.

6.3.2. Estrategia de distribución por pilas general

Básicamente la estrategia funciona de un forma muy parecida a la estrategia anterior (Ver 6.3.1) pero se diferencia en el hecho de que la estrategia se aplica sobre todas las pilas al mismo tiempo y no como la anterior que esperaba a que una pila estuviese completamente estibada para continuar con la siguiente (Ver figura 6.2).

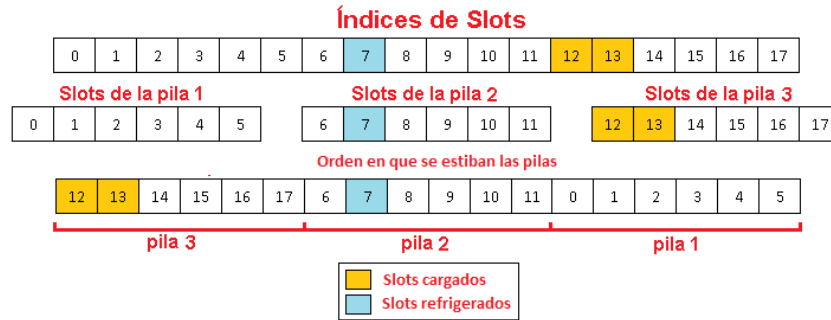


Figura 6.2: Índices por pila general

6.3.3. Estrategia de distribución por niveles de cada pila

La estrategia de distribución por niveles consiste en ordenar los índices de las ranuras pertenecientes a los niveles, del nivel inferior al nivel superior, de cada una de las pilas (Ver figura 6.3).

Después de ordenado por niveles, se asignan las variables P, L, W, H y S de la misma forma en que se asignan para la estrategia de distribución por pila.

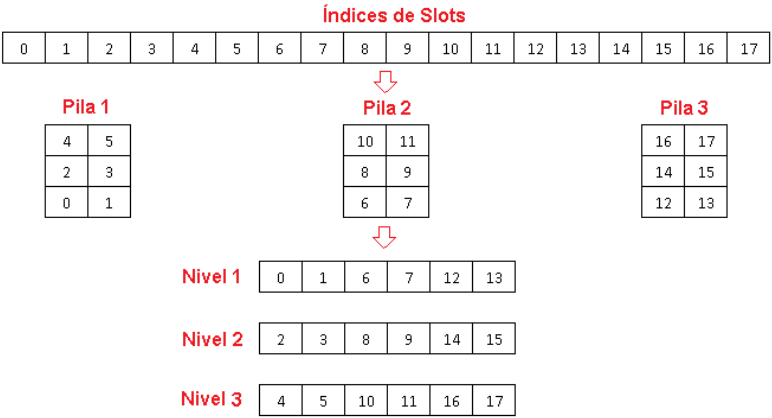


Figura 6.3: Índices por niveles

Capítulo 7

Modelo extendido

Este modelo se presenta como una extensión al modelo presentado en el capítulo 6. Se extiende el modelo proponiendo la idea de utilizar ranuras y contenedores virtuales. Las ranuras virtuales se utilizan para que los contenedores reales puedan ser siempre estibados y los contenedores virtuales se utilizan para indicar cuando una ranura no fue estibada por un contenedor real. La cantidad de contenedores y ranuras, virtuales y no virtuales, deben tener la misma cardinalidad. Adicionalmente, los contenedores virtuales tienen características como el largo, ancho, alto y puerto de descarga iguales a 0.

7.1. Conjunto de constantes adicionales

A continuación, se presenta un conjunto de constantes adicionales que sólo son utilizadas en este modelo (Ver cuadro 7.1).

Adicionalmente, para este modelo no se utiliza la constante $Cont^V$

7.2. Variables de decisión

Debido a la utilización de contenedores y ranuras virtuales, es necesario crear, eliminar y modificar algunas variables de decisión. (Ver cuadro 7.2).

Las variables de decisión utilizadas para calcular el costo de las soluciones encontradas, son iguales a las del modelo sin extender.

7.2.1. Restricciones del modelo

Para un correcto funcionamiento del modelo extendido, es necesario replantear el modelo de restricciones.

Índices y constantes	Descripción
SlotsV	Subconjunto de Slots que indica los espacios virtuales, este conjunto tiene el tamaño de $Cont^{Load}$.
SlotsR	Subconjunto de Slots que indica los espacios reales.
ContV	Subconjuntos de Cont que indica los contenedores virtuales, este conjunto tiene el tamaño de $SlotsR$.
ContR	Subconjuntos de Cont que indican los contenedores reales, cargados y no cargados.
$OverCont$	Conjunto de variables booleanas del tamaño $ Slots $, donde cada elemento representa la relación $OverCont_i = (i \geq SlotsR)$
CT	Conjunto de tipos de contenedores.
SCT^i	Conjunto de contenedores de tipo i .

Cuadro 7.1: Tabla de constantes adicionales.

Restricciones no modificadas

Debido a que es una extensión del modelo anterior, existen varias restricciones que satisfacen los nuevos conceptos del modelo, éstas son:

- De la restricción 6.2 a la 6.10.
- De la restricción 6.15 a la 6.30.
- De la restricción 6.34 a la 6.35.

Estas restricciones se aplican en este modelo extendido, de la misma forma en que se aplican para el modelo original.

Eliminación de restricciones

Para poder dar soporte al modelo extendido, se eliminan las siguientes restricciones:

- La restricción 6.1.
- De la restricción 6.11 a la 6.14.
- De la restricción 6.31 a la 6.33.

Estas restricciones se eliminan porque provocarían resultados indeseados en el modelo extendido.

Variables de decisión	Descripción
$RC = \{RC_1, \dots, RC_{ ContLoad }\}$	$RC_i \in Slots$, índice del espacio o ranura de un contenedor real estibado.
$C = \{C_1, \dots, C_{ Slots }\}$	$C_i \in Slots$, índice del espacio o ranura en el que el contenedor será estibado.
$S = \{S_1, \dots, S_{ Slots }\}$	$S_i \in Cont$, índice del contenedor, real o virtual, estibado.
$L = \{L_1, \dots, L_{ Slots }\}$	$L_i \in Length$, largo del contenedor estibado en $Slots_i$.
$H = \{H_1, \dots, H_{ Slots }\}$	$H_i \in Height$, alto del contenedor estibado en $Slots_i$.
$W = \{W_1, \dots, W_{ Slots }\}$	$W_i \in Weight$, peso del contenedor estibado en $Slots_i$.
$WD = \{WD_1, \dots, WD_{ Slots }\}$	$WD_i \in Weight$, peso del contenedor estibado en $Slots_i$. (Arreglo de variables no escalares).
$P = \{P_1, \dots, P_{ Slots }\}$	$P_i \in POD$, puerto de descarga del contenedor estibado en $Slots_i$.
$NRSR$ $\{NRSR_1, \dots, NRSR_{ Slots^R }\}$	$NRSR_i \in \{0, 1\}$, indica si se carga un contenedor no refrigerado en una ranura o espacio refrigerado.
$HS = \{HS_1, \dots, HS_{ Stack }\}$	$HS_i \in Stack$, límite de altura por pila.
$OPT = \{OPT_1, \dots, OPT_{ Stack }\}$	$OPT_i \in Cont_K$, número de contenedores diferentes en la pila i .
$WT = \{WT_1, \dots, WT_{ Stack }\}$	$WT_i \in \{0, Stack_i^W\}$, peso total de los contenedores estibados en la pila i .
$GCenY$ $\{GCenY_{1,1}, \dots, GCenY_{ Stack Slots }\}$	$GCenY_{i,s} \in \{0, maxLevel(i)\}$, centro de gravedad en eje Y para la pila i y el slot s .
$GCV = \{GCV_1, \dots, GCV_{ Stack }\}$	$GCV_i \in \{0, maxLevel(i)\}$, centro de gravedad en el eje Y.
$GCD = \{GCD_1, \dots, GCD_{ Stack }\}$	$GCD_i \in \{0, maxLevel(i)\}$, distancia al centro de gravedad ideal por pila.
$OUT = \{OUT_1, \dots, OUT_{ Stack }\}$	$OUT_i \in Cont^{\{20,40\}}$, número de contenedores i de 20 y 40 pies en la pila k .
$OUTE$	$0 \leq OUTE \leq Stack $, número de pilas vacías.
$VSC = \{VSC_1, \dots, VSC_{ SlotsV }\}$	$VSC_i \in \{0, 1\}$, indica verdadero si se ha estibado un contenedor no virtual en un espacio virtual, de lo contrario indica falso.
$VC = \{VC_1, \dots, VC_{ ContV }\}$	$VC_i \in ContV$, contenedores virtuales.
$VS = \{VS_1, \dots, VS_{ SlotsV }\}$	$VS_i \in SlotsV$, ranuras virtuales.
CS	$0 \leq CS \leq Slots $, número de contenedores estibados.

Cuadro 7.2: Tabla de variables de decisión.

Nuevas restricciones implementadas

A continuación, se muestran las nuevas restricciones adicionadas al modelo.

$$channel(S, C) \quad (7.1)$$

$$element(C_i, VSC_i, OverCont) \quad \forall i \in ContR \wedge i \notin Cont_F^{40} \quad (7.2)$$

$$VSC_i == 0 \quad \forall i \in Cont_F^{40} \quad (7.3)$$

$$C_{i+1} = C_i + 1 \quad \forall i \in Cont_A^{40} \quad (7.4)$$

$$VC_i = C_{getVC(i)} \quad \forall i \in ContV \quad (7.5)$$

$$VC_i < VC_{i+1} < \dots < VC_{i+n} \quad i = 0, n = |ContV| \quad (7.6)$$

$$VS_i = S_{getVS(i)} \quad \forall i \in SlotsV \quad (7.7)$$

$$VS_i < VS_{i+1} < \dots < VS_{i+n} \quad i = 0, n = |SlotsV| \quad (7.8)$$

$$exactly(W, 0, |VC|) \quad (7.9)$$

$$exactly(P, 0, |VC|) \quad (7.10)$$

$$exactly(H, 0, |VC|) \quad (7.11)$$

$$exactly(L, 0, |VC|) \quad (7.12)$$

$$exactly(W, i, ContEW_i) \quad \forall i \in ContEW \quad (7.13)$$

$$exactly(P, i, ContEP_i) \quad \forall i \in ContEP \quad (7.14)$$

$$exactly(H, i, ContEH_i) \quad \forall i \in ContEH \quad (7.15)$$

$$exactly(L, i, ContEL_i) \quad \forall i \in ContEL \quad (7.16)$$

$$SCT_j^i \geq SCT_{j+1}^i \geq \dots \geq SCT_{j+n}^i \quad \forall i \in CT | j = 0, n = |SCT^i| \quad (7.17)$$

$$CS = ContR - OCNS \quad (7.18)$$

La restricción (7.1) relaciona los contenedores con los espacios o ranuras. Las restricciones (7.2) y (7.3) se utilizan para identificar la cantidad de contenedores estibados en espacios virtuales. La restricción (7.4) es para asegurar que los dos contenedores de 20 pies, utilizados para representar uno de 40 pies, queden estibados de forma continua. La restricción (7.5) tiene como objetivo agrupar en la variable VC los contenedores virtuales y la restricción (7.6) es utilizada para romper la simetría que existe entre los contenedores virtuales, ya que estibar un contenedore virtual u otro contenedor virtual en un espacio o ranura i , no debería afectar el resultado. Las restricciones (7.7) y (7.8) funcionan de la misma forma que las restricciones (7.5) y (7.6) pero con la diferencia de que las restricciones se aplican sobre variables que representan las ranuras virtuales. Las restricciones (7.9), (7.10), (7.11) y (7.12) son usadas para especificar en las variables W , P , H y L la cantidad de contenedores virtuales que deben de existir. Las restricciones (7.13), (7.14), (7.15) y (7.16) tienen como objetivo limitar el número de contenedores reales a estibar de acuerdo a características como al peso, el puerto de descarga, la altura y el largo de los contenedores. La restricción (7.17) es para romper la simetría que existe entre los contenedores que son del mismo tipo. La restricción (7.18) es utilizada para obtener la cantidad de contenedores reales estibados.

En el cálculo de las variables que representan el costo de la soluciones, el único cambio que se efectuó entre el modelo original y el modelo extendido fue el cálculo de la variable $OCNS$. Por lo anterior, se remplace la restricción (6.40) por la restricción (7.19) que sirve para contar el costo de contenedores no estibados en el buque.

$$OCNS = \sum_{i \in Stacks} VSC_i \quad (7.19)$$

7.3. Estrategia de distribución

Para este modelo se utilizan dos estrategias de distribución. En ambas estrategias se hace uso de las variables auxiliares P, L, W y H . Las estrategias de distribución son similares a las del anterior modelo. Además, también se tiene en cuenta que la prioridad para distribuir las variables auxiliares P, L, W y H se basa en la cantidad de valores diferentes en el dominio de dichas variables.

7.3.1. Estrategia de distribución por pila específica

Esta estrategia funciona agrupando las pilas de la misma forma en la que se hace para el modelo original (Ver 6.3.1). La única diferencia radica en que para este modelo no se tiene en cuenta los espacios o ranuras virtuales para hacer la distribución (Ver figura 7.1).

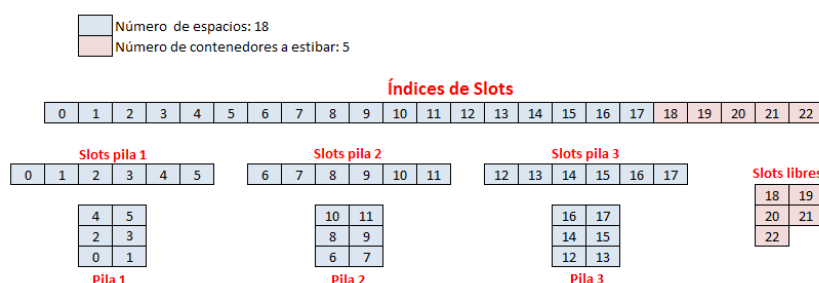


Figura 7.1: Índices por pila

7.3.2. Estrategia de distribución por niveles de cada pila

La estrategia de distribución por niveles consiste en ordenar los índices de las ranuras pertenecientes a los niveles, del nivel inferior al nivel superior, de cada una de las pilas, de la misma forma en la que se hace para el modelo original (Ver 6.3.2). A diferencia del modelo original, no se tiene en cuenta los espacios o ranuras virtuales para hacer la distribución (Ver figura 7.2).

Las variables P, L, W y H se distribuyen después de que se han ordenado y posteriormente, se distribuye sobre la variable S , de la misma forma en que se distribuye para las variables auxiliares, para encontrar soluciones factibles.

7.3.3. Estrategia de distribución por niveles de cada pila y por contenedores reales

Básicamente esta estrategia funciona de forma similar a la estrategia anterior (Ver 7.3.2), pero se diferencia en que al final no se distribuye sobre la variable S , si no que lo hace sobre la variable RC , asignado el mínimo valor encontrado. Esta

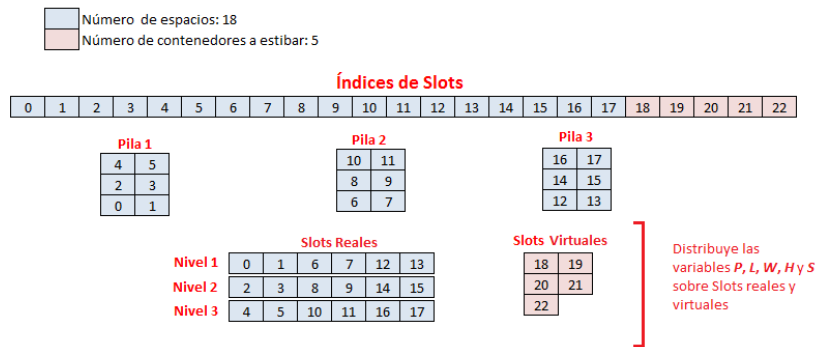


Figura 7.2: Índices por niveles

estrategia encuentra resultados debido a que si se estiban todos los contenedores reales, las restricciones desde (7.5) hasta (7.12) se encargarían de asignar todos los contenedores virtuales faltantes.

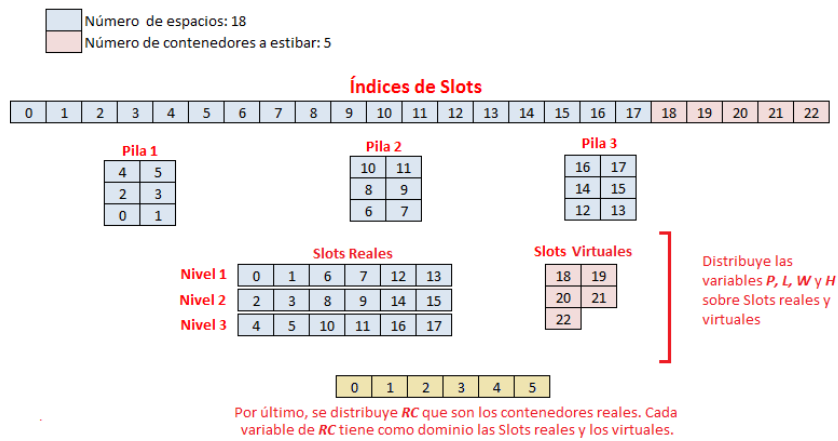


Figura 7.3: Índices por niveles y contenedores reales

Capítulo 8

Implementación de modelos.

La idea de implementar los modelos se realiza con el fin de comprobar la efectividad y la veracidad de los modelos creados. Las implementaciones son sometidas a pruebas con instancias reales del problema **CSPADL** y los resultados obtenidos se exponen en la sección 9.

La implementación del modelo original y el modelo extendido, presentados en éste documento, se realiza sobre el lenguaje de programación Gecode, debido a que es un entorno de programación bastante ideal para la construcción de aplicaciones fundamentadas en la optimización de procesos por medio de restricciones.

La implementación sobre Gecode se realizó con el objetivo de crear una librería que pudiese ser utilizada desde otros programas externos. La librería recibe un archivo con un formato determinado, en el que se especifica la lista de puertos de descarga, la lista de locaciones del barco, la lista de contenedores, la lista de pilas y la lista de celdas del barco.

La estructura del archivo de entrada se muestra en la figura 8.1. A continuación, se explicará la estructura y lo que significa cada una de las variables del archivo de entrada:

1. Especificación general del archivo de entrada en la primera línea.
 - NumPortsDischarge: Número de puertos de descarga.
 - NumContainerLoad: Número de contenedores a cargar.
 - NumContainerLoaded: Número de contenedores cargados.
 - NumStacks: Número de pilas.
 - NumCell: Número de celdas.
 - NumLocations: Número de locaciones.
 - NumTiers: Número de niveles.

2. Puertos de descarga (#POD).
 - Lista de identificadores de puertos de descarga.
3. Locaciones del barco (#LOCATIONS).
 - Lista de identificadores de locaciones del barco.
4. Lista de contenedores a estibar o ya estibados (#CONTAINERS_TOLOAD, #CONTAINERS_TOLOADED).
 - StackId: Identificador de la pila, es 0 cuando el contenedor no está estibado.
 - CellId: Identificador de la celda, es 0 cuando el contenedor no está estibado.
 - numPosition: Especifica la posición en la celda en la que está estibado el contenedor, solo aplica cuando el contenedor está estibado.
 - numWeigth: Peso del contenedor.
 - numHeigth: Altura del contenedor.
 - numLength: Largo del contenedor.
 - NumPortDischarge: Identificador del puerto de descarga.
 - isReefer: Especifica si un contenedor es refrigerado.
 - numLocation: Especifica la locación en la que se debe estibar ó está estibado el contenedor.
5. Lista de pilas (#STACKS)
 - maxWeigth: Máximo peso de la pila.
 - maxHeigth: Máxima altura de la pila.
 - numLocation: Locación de la pila.
6. Lista de celdas (#CELLS).
 - StackId: Identificador de la pila.
 - IsReeferFore: Es una celda refrigerada en la popa.
 - IsReeferAft: Es una celda refrigerada en la proa.
 - CapFore: Representa cuando la popa de la celda está disponible.
 - CapAft: Representa cuando la proa de la celda está disponible.
 - Cap40: Representa cuando la celda está disponible para contenedores de 40 pies.
 - numLocation: Identificador de la locación.

```

INPUT FILE

NumPortsDischarge NumContainerLoad NumContainerLoaded NumStacks
NumCell NumLocations NumTiers

#POD
ListPortsDischarges

#LOCATIONS
ListLocations

#CONTAINERS_TOLOAD
StackId CellId numPosition numWeigth numHeigth numLength NumPortDischarge
isReefer numLocation
...
...

StackId CellId numPosition numWeigth numHeigth numLength NumPortDischarge
isReefer numLocation

#CONTAINERS_TOLOADED
StackId CellId numPosition numWeigth numHeigth numLength NumPortDischarge
isReefer numLocation
...
...

StackId CellId numPosition numWeigth numHeigth numLength NumPortDischarge
isReefer numLocation

#STACKS
maxWeigth maxHeigth numLocation
...
...

maxWeigth maxHeigth numLocation

#CELLS
StackId isReeferFore isReeferAft CapFore CapAft Cap40 numLocation
...
...

StackId isReeferFore isReeferAft CapFore CapAft Cap40 numLocation

```

Figura 8.1: Archivo de entrada.

La salida de las implementaciones es un archivo con un formato específico que contiene la solución al problema, la estructura del archivo se muestra en la figura 8.2. A continuación, se explica la estructura del archivo de salida.

1. Datos generales de la solución:

- O: Costo total de la solución.

- OGCTD: Costo de la sumatoria de los pesos otorgados por la distancia al centro de gravedad ideal de cada pila.
 - OR: Costo del número de contenedores no refrigerados, estibados en ranuras refrigeradas.
 - OP: Costo del número de contenedores diferentes por cada pila.
 - OU: Costo por número de pilas usadas.
 - OCNS: Costo de número de contenedores no estibados.
 - TimeSol: Tiempo en que encuentra última solución.
 - TimeFin: Tiempo total de ejecución.
 - FileIn: Nombre de la instancia procesada.
 - Model: Modelo utilizado.
 - Strategy: Estrategia de distribución utilizada.
2. Índices de contenedores indexados por los índices de los espacios del buque (#S).
- ListS: Muestra una lista de números separados por coma que representan lo siguiente $S = \{S_1, \dots, S_{|Slots|}\}$ tal que $S_i \in \text{Cont}$, índice del contenedor en el $Slot_i$.

OUTPUT FILE

O OGCTD OR OP OU OCNS TimeSol TimeFin FileIn Model Strategy

#S

ListS

Figura 8.2: Archivo de salida.

Capítulo 9

Resultados de los modelos

En este capítulo se muestran los resultados obtenidos por cada una de las estrategias de distribución del modelo original y del modelo extendido propuesto en este documento. Las pruebas realizadas sobre cada una de las estrategias de distribución se hacen sobre datos reales. Adicionalmente, para estas pruebas, se utiliza una maquina con sistema operativo Linux con 8 procesadores a 2.5 GHz y 4 GB de memoria.

9.1. Clasificación de instancias a probar

Se realiza una clasificación de instancias con el fin de probar con mayor precisión el comportamiento de los dos modelos. Las instancias se agrupan de acuerdo a las diferentes características que éstas posean. El cuadro 9.1 muestra la categorización de las instancias a probar. Cada registro del cuadro 9.1 muestra el **(Grp.)**, que simplemente es una secuencia para clasificar un grupo de instancias; la columna **(#Inst.)**, indica el número de instancias por grupo; la columna **(Cap.)**, muestra la capacidad mínima, la máxima y el promedio en las locaciones del barco en TEUs; la columna **(Cont.)**, muestra el mínimo, el máximo y el promedio de contenedores a estibar en TEUs; Los campos **(20')**, **(40')**, **(R)** y **(HC)** son características de los contenedores por las que se agrupan las instancias; la columna **(#POD)**, indica el número de instancias del grupo que contienen 1, 2 o más de 3 puertos de descarga para los contenedores.

En total son 234 instancias clasificadas en diferentes grupos. Esta clasificación permite mostrar de forma particular, como responden los modelos con sus diferentes distribuidores ante diferentes tipos de instancias.

Grp.	#Inst.	Cap. (TEUs)			Cont. (TEUs)			20'	40'	R	HC	#POD		
		<i>min</i>	<i>max</i>	<i>prom</i>	<i>min</i>	<i>max</i>	<i>prom</i>					1	2	≥ 3
1	24	8	168	69	8	136	54	*				24		
2	2	124	130	127	78	130	104	*					2	
3	12	30	124	73	8	124	67	*	*			10	2	
4	50	38	220	110	8	202	83	*	*		*	38	10	2
5	2	156	172	164	6	10	8	*	*	*		2		
6	8	72	176	128	8	170	95	*	*	*	*	8		
7	4	48	176	86	9	148	51	*	*	*	*		1	3
8	16	16	124	64	8	124	55		*			16	1	
9	81	6	208	81	2	202	62		*		*	81		
10	12	40	148	83	10	136	67		*		*		12	
11	2	42	130	86	10	42	26		*	*		1	1	
12	13	42	176	86	8	150	51		*	*	*	13		
13	6	72	176	130	10	156	75		*	*	*		6	
14	2	90	126	108	56	92	74		*	*	*			2

Cuadro 9.1: Tabla de clasificación de instancias

9.2. Resultados de los modelos

En esta sección se pretende mostrar los resultados obtenidos por cada una de las estrategia de distribución realizadas en los 2 modelos. Primero se muestra el porcentaje de éxito buscando soluciones a las instancias. Posteriormente, se muestra una sección en la que se especifica la calidad de las soluciones encontradas.

Porcentaje de éxito

A continuación, se muestran el porcentaje de éxito que tuvieron las estrategias de distribución de cada uno de los modelos. El cuadro 9.2 muestra los porcentajes de éxito de cada estrategia de distribución para cada uno de los grupos que indican una determinada clasificación de instancias. El cuadro 9.2 tiene la columna (**Grp.**) que indica el grupo de instancias probadas; las columnas **Modelo Original** y **Modelo Extendido**, que se encuentran separadas por 3 columnas que indican las 3 estrategias de distribución implementadas para el modelo original y el modelo extendido (**Estr. 1**, **Estr. 2** y **Estr. 3**). Adicionalmente, cada estrategia esta separada por 2 columnas que indican el porcentaje de éxito encontrando buenas soluciones y las óptimas soluciones (**Sol** y **Opt**).

Los experimentos se realizan con un tiempo límite de 1 minuto por instancia. En la tabla 9.2 se muestra que se encuentra una solución óptima para una instancia sólo cuando el proceso de búsqueda de soluciones para dicha instancia termina antes del tiempo limite. Por lo contrario, si la búsqueda de soluciones para la instancia supera el límite de tiempo, se puede decir que la solución no es óptima pero si satisfactible, si ha encontrado al menos una.

En los experimentos realizados, se puede denotar con facilidad que el modelo original tiene un mejor rendimiento con respecto al modelo extendido. Lo anterior,

Grp.	Modelo Original (%)						Modelo Extendido (%)					
	Estr. 1		Estr. 2		Estr. 3		Estr. 1		Estr. 2		Estr. 3	
	Sol	Opt	Sol	Opt	Sol	Opt	Sol	Opt	Sol	Opt	Sol	Opt
1	100	79	100	79	100	75	92	67	100	71	88	58
2	100	100	100	100	100	100	100	50	100	50	50	50
3	100	75	100	75	100	58	92	58	92	50	92	42
4	94	34	88	34	88	26	70	20	74	26	70	24
5	100	100	100	100	100	100	100	100	50	50	50	0
6	100	25	88	13	88	13	50	13	63	13	63	13
7	100	50	100	50	100	50	75	50	50	50	50	50
8	100	94	100	94	100	94	88	88	100	94	88	81
9	100	78	98	77	100	70	93	67	98	64	90	48
10	100	25	92	25	92	17	100	25	92	17	83	17
11	100	100	100	100	100	100	100	50	100	50	100	50
12	100	46	100	46	100	38	92	31	100	23	100	23
13	100	17	100	17	100	17	33	17	100	17	100	17
14	100	0	100	0	100	0	0	0	100	0	100	0
Tot.	99	61	96	60	97	54	84	50	90	49	84	40

Cuadro 9.2: Tabla de porcentajes de éxito

se expresa con respecto a los resultados obtenidos en los experimentos, en los cuales se observa que las estrategias de distribución del modelo original presentan un mayor porcentaje de éxito con respecto a las estrategias del modelo extendido.

Sobre el modelo original se puede denotar que la primera estrategia, distribución por pila específica (ver 6.3.1), fue la que obtuvo mayor cantidad de soluciones, puesto que pudo resolver el 99 % de todas las instancias y obtuvo un 61 % de óptimas soluciones. La segunda estrategia, distribución por pila general (ver 6.3.2), obtuvo un porcentaje de éxito del 96 % buscando soluciones y ésta estrategia obtuvo un porcentaje del 60 % encontrando soluciones óptimas. Por último, la tercera estrategia, distribución por niveles de cada pila (ver 6.3.3), obtuvo un porcentaje de éxito encontrando soluciones de 97 % y un 54 % encontrando soluciones óptimas.

En el modelo extendido, la primera estrategia, distribución por pila específica (ver 7.3.1), obtiene un porcentaje de éxito del 84 % encontrando soluciones y un 50 % encontrando soluciones óptimas. La segunda estrategia, distribución por niveles de cada pila (ver 7.3.2), es la mejor estrategia en éste modelo puesto que obtiene un porcentaje de éxito del 90 % encontrando soluciones factibles y un 49 % encontrando soluciones óptimas. Por último, la tercera estrategia, distribución por niveles de cada pila y contenedores reales (ver 7.3.3), es la que menor porcentaje de éxito obtiene, ésta encuentra un porcentaje del 84 % encontrando soluciones y un 40 % encontrando soluciones óptimas.

Calidad de las soluciones

En los resultados obtenidos se puede evidenciar la calidad de soluciones encontradas para las instancias del problema **CSPADL**. En la sección anterior (Ver 9.2), se puede ver que los porcentajes de éxito se encuentran divididos en soluciones normales, es decir, que no se puede asegurar que sean óptimas, y soluciones óptimas. Lógicamente, una solución óptima es la mejor solución encontrada para la instancia y no es necesario demostrar su calidad. Sin embargo, es necesario mostrar el tiempo en que se encuentran dichas soluciones óptimas. En éste documento se muestra el tiempo que le toma a las estrategias de cada modelo encontrar la solución óptima (ver 9.1).

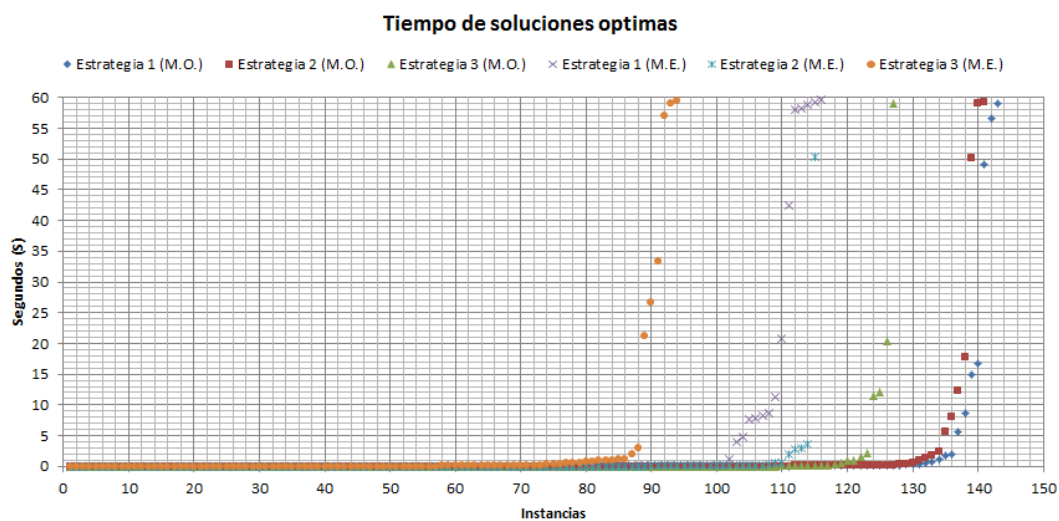


Figura 9.1: Tiempo de soluciones óptimas. Las siglas (M.O.) y (M.E.) hacen referencia al modelo original y al modelo extendido.

En el diagrama 9.1 se ordenan, de menor a mayor, las instancias de acuerdo al tiempo utilizado por las estrategias para obtener la solución más óptima de una instancia determinada del problema. Se puede ver con claridad que el 93 %, el 92 % y el 94 % de las instancia resueltas con la primera, segunda y tercera estrategia del modelo original se resuelven en menos de 1 segundo y que el 87 %, 95 % y 86 % de las instancias resueltas con la primera, segunda y tercera estrategia del modelo extendido tambien se resuelven en menos de 1 segundo.

Por otro lado, las soluciones que no son óptimas es necesario evaluarlas y saber que tan buena calidad poseen las respuestas encontradas. Para mostrar la calidad de las soluciones se presentan dos diagramas de dispersión para cada una de las estrategias del modelo original y el modelo extendido. Los diagramas se construyen con 44 instancias que no se pudieron resolver en el tiempo límite en todas las estrategias de distribución. El primer diagrama muestra los porcentajes de fallo por cada uno de los objetivos buscados al momento de resolver las instancias. El segundo diagra-

ma muestra los tiempos en los que se encontraron las últimas soluciones para las instancias.

En el primer diagrama de cada una de las estrategias de distribución, del modelo original y extendido, se muestra los siguientes datos:

1. Porcentaje de contenedores no estibados (PCNE).

$$PCNE = \frac{OCNS}{|Cont|} * 100 \quad (9.1)$$

La fórmula representa el subconjunto de contenedores que no pudieron ser estibados en el buque. En el modelo extendido la variable *Cont* es remplazada por *ContR*.

2. Porcentaje de pilas con diferente puerto de descarga (PPDP).

$$PPDP = \frac{OP}{|Stack| * |POD|} * 100 \quad (9.2)$$

En la ecuación se muestra la sumatoria de la cantidad de puertos de descarga diferentes en cada una de las pilas y éste valor se divide por el máximo número de puertos de descarga por pila posibles.

3. Porcentaje de pilas adicionales utilizadas (PPAU).

$$PPAU = \frac{OU}{|Stack|} * 100 \quad (9.3)$$

Lo anterior ecuación muestra la cantidad de pilas adicionales que se necesitaron para estibar los contenedores.

4. Porcentaje de contenedores no refrigerados estibados en ranuras refrigeradas (PCRR).

$$PCRR = \frac{OR}{|Cont|} * 100 \quad (9.4)$$

La fórmula anterior muestra el subconjunto de contenedores no refrigerados que se estibarón en ranuras refrigeradas. En el modelo extendido la variable *Cont* es remplazada por *ContR*.

5. Porcentaje de distancia del centro de gravedad ideal (PDCG).

$$PDCG = \frac{OGCTD}{GCSY} \quad (9.5)$$

La fórmula muestra el porcentaje de la distancia entre el centro de gravedad ideal y el centro de gravedad obtenido para cada una de las pilas.

En el segundo diagrama de cada estrategia, del modelo original y extendido, muestra para cada una de las 44 instancias probadas, el tiempo en que se encontró la última solución para una determinada instancia. Este tiempo diagramado corresponde al tiempo utilizado para encontrar los objetivos mostrados en el primer diagrama.

Diagramas del modelo original

En la figuras 9.2, 9.3 y 9.4 se muestran los diagramas de dispersión que visualizan los resultados obtenidos utilizando la primera, la segunda y la tercera estrategia de distribución del modelo original (ver 6.3).

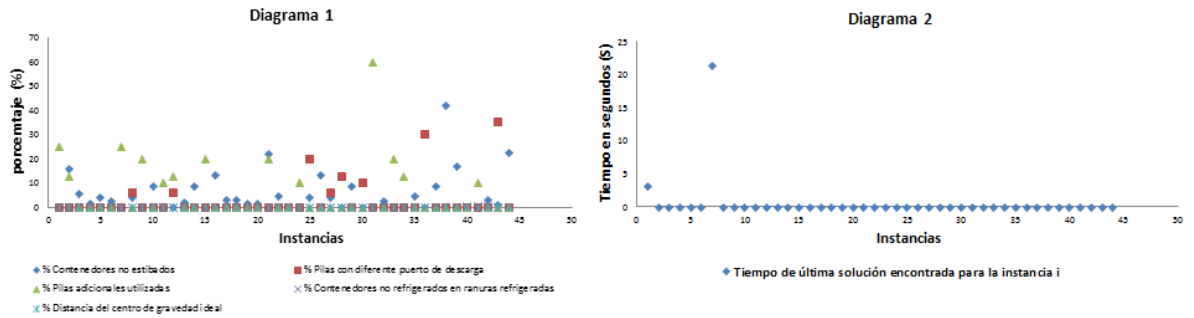


Figura 9.2: Diagramas de dispersión estrategia No.1 - Modelo original

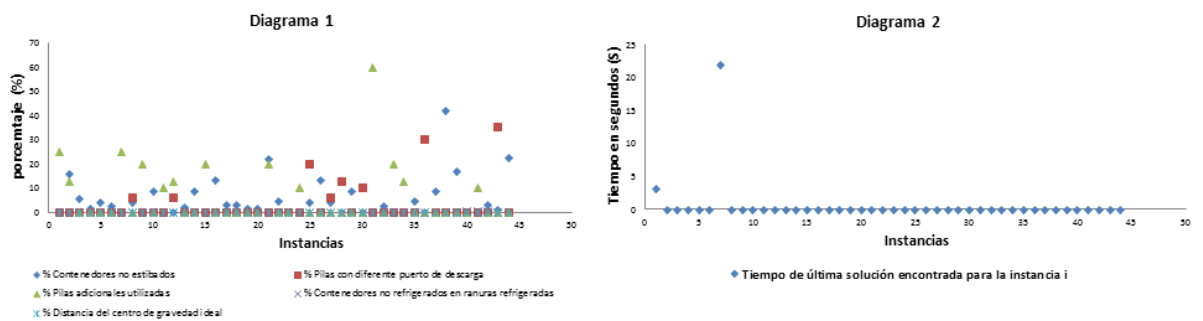


Figura 9.3: Diagramas de dispersión estrategia No.2 - Modelo original

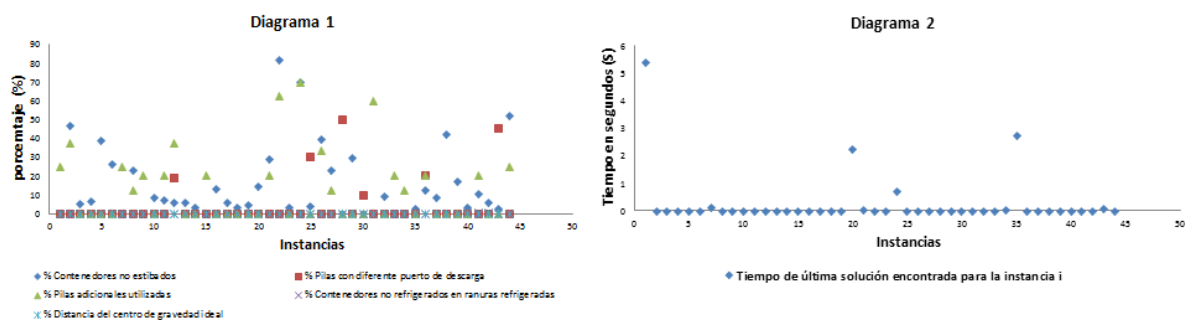


Figura 9.4: Diagramas de dispersión estrategia No.3 - Modelo original

Se puede denotar que los resultados obtenidos con la primera y segunda estrategia del modelo original son similares tanto en el tiempo, en que se encuentra la solución, como en el porcentaje de objetivos fallidos; se podría deducir que obtiene

resultados similares debido a que las estrategias se basan en la distribución por pila. Adicionalmente, se puede observar que en éstas dos estrategias los objetivos fallidos están por debajo del 25 % para el 91 % de las instancias. Lo anterior, implica el hecho de que las estrategias a pesar de que no pudieron resolver el problema en el tiempo límite, se puede decir que las soluciones encontradas son muy buenas, si es que no son las óptimas. En los diagramas de la tercera estrategia de distribución del modelo original se puede observar que para un 91 % de las instancias probadas se encuentra que los objetivos fallidos están por debajo del 50 %. Se puede denotar con claridad que los resultados obtenidos por las tres estrategias de distribución del modelo original obtienen su última solución en menos de 1 segundo.

Diagramas del modelo extendido

En la figuras 9.5, 9.6 y 9.7 se muestran los diagramas de dispersión que visualizan los resultados obtenidos utilizando la primera, la segunda y la tercera estrategia de distribución del modelo extendido (ver 7.3).

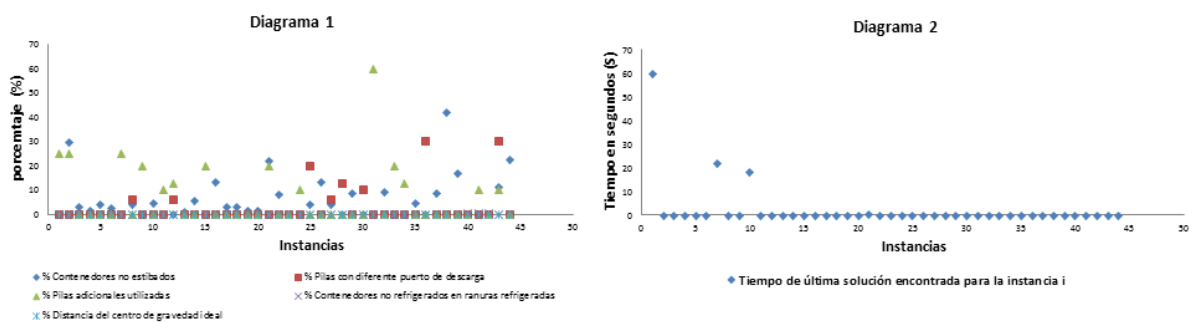


Figura 9.5: Diagramas de dispersión estrategia No.1 - Modelo extendido

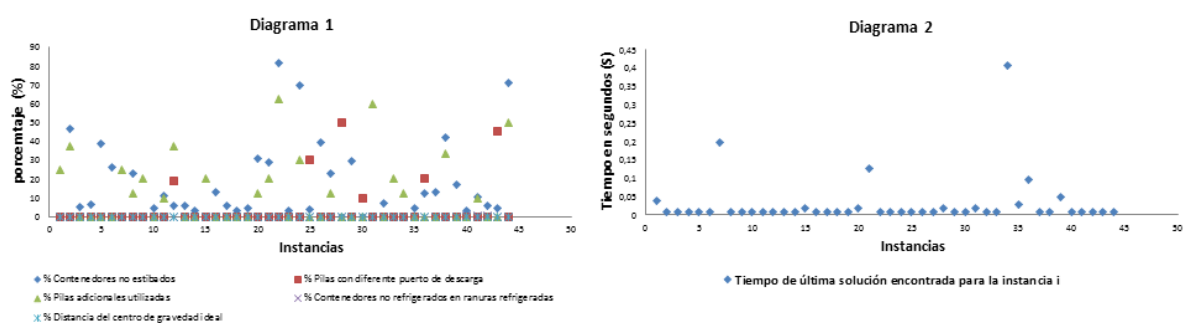


Figura 9.6: Diagramas de dispersión estrategia No.2 - Modelo extendido

Las respuestas obtenidas por la primera estrategia de distribución del modelo extendido muestra que para un 95 % de las instancias probadas los objetivos fallidos se encuentra por debajo del 32 %. La estrategia de éste modelo obtiene valores similares

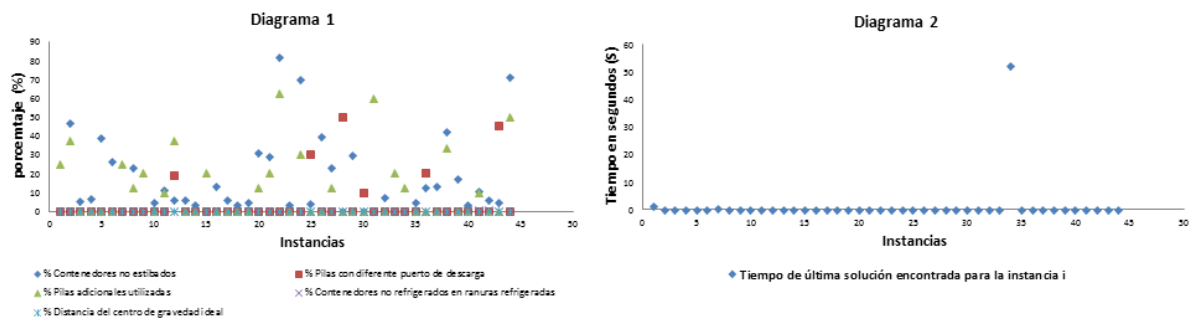


Figura 9.7: Diagramas de dispersión estrategia No.3 - Modelo extendido

a los obtenidos por la primera y segunda estrategia del modelo original. La segunda y tercera estrategia del modelo extendido obtuvo valores similares a los que se obtuvieron con la tercera estrategia del modelo original. Igual que en la tercera estrategia los objetivos fallidos están por debajo del 50 %. Ésta similitud que se encuentra en los datos obtenidos por éstas estrategias se debe a que las distribuciones se basan en distribuir primero las variables por los niveles, del inferior al superior, de cada pila.

Capítulo 10

Conclusiones

- La industria del transporte marítimo de mercancías a través de buques porta-contenedores posee un gran impacto en la sociedad y economía mundial. Es bien concebida la idea de modelar software capaz de simular ciertos comportamiento de esta industria, de esta manera se aporta un conocimiento previo para que las personas que trabajan en éste campo se desenvuelvan mucho mejor.
- Los modelos propuestos en este documento representan una abstracción del problema real que la industria marítima afronta.
- En éste documento se trata específicamente el problema **CSPADL** que es la organización de contenedores en secciones de bahía que se encuentran por encima de la cubierta del barco. El problema **CSPADL** en computación es denominado un problema NP-Completo y hace parte de una de las descomposiciones del problema de estibar contenedores en un buque. Sobre éste problema se desarrolla un modelo en restricciones, denominado modelo original, con tres estrategias de distribución diferentes que obtienen respuesta factibles en un 99 %, un 96 % y un 97 % de las 234 instancias probadas, y de las cuales el 61 %, el 60 % y el 54 % de los resultados obtenidos, al probar las 234 instancias, son soluciones óptimas. Los anteriores resultados fueron obtenidos al probar las instancias con un límite de tiempo de 1 minuto.
- Las tres estrategias de distribución del modelo original obtienen soluciones en menos de 1 segundo para un 93 %, 92 % y 94 % de las instancias resueltas óptimamente.
- Investigando si se podían obtener mejores respuestas con el modelo original al problema **CSPADL** se crea el modelo extendido con tres estrategias de distribución, las cuales fueron adaptadas del modelo original. Sin embargo, los resultados obtenidos por el modelo extendido no fueron mejores que los del

modelo original, puesto que éste obtuvo para las mismas 234 instancias y con las tres estrategias de distribución del modelo extendió un porcentaje de éxito del 84 %, el 90 % y el 84 %. Donde las tres estrategias del modelo extendido obtuvieron sobre las 234 instancias un 50 %, un 49 % y un 40 % respuestas que se pueden asegurar como óptimas.

- Las tres estrategias de distribución del modelo extendido obtienen soluciones en menos de 1 segundo para un 87 %, 95 % y 86 % de las instancias resueltas óptimamente.
- Las respuestas que se obtuvieron con el modelo original y que no se pueden asegurar como respuestas óptimas; son muy buenos resultados, teniendo en cuenta que todos los objetivos fallan en un porcentaje menor al 25 % en 2 de las 3 estrategias de distribución implementadas para el modelo original.
- Las respuestas que se obtuvieron con el modelo extendido y que no se pueden asegurar como respuestas óptimas; no son resultados tan buenos como los obtenidos por el modelo original, teniendo en cuenta que todos los objetivos fallan en un porcentaje menor al 32 % para la mejor estrategia de distribución implementada en el modelo extendido.
- Se nota con claridad que el modelo original es capaz de probar optimalidad en un tiempo límite para la mayoría de las instancias mientras que el modelo extendido en algunas ocasiones encuentra la solución óptima más rápido que el modelo original pero requiere más tiempo para probar optimalidad en la mayoría de las instancias.
- Las respuestas obtenidas por el modelo original y el extendido difieren en los cambios que se realizan para pasar de un modelo a otro. Primero, la cantidad de variables de decisión del modelo original es menor a la cantidad de variables de decisión utilizadas por el modelo extendido, lo anterior, sucede porque en el modelo extendido se crean variables de decisión para modelar los contenedores y ranuras virtuales, mientras que en el modelo original sólo se utiliza una variable de decisión para representar los contenedores virtuales y no se modelan ranuras virtuales. Segundo, en el modelo extendido, al generar contenedores y ranuras virtuales es necesario crear restricciones, no utilizadas en el modelo original, para quitar del espacio de búsqueda las soluciones simétricas (ver restricciones 7.5, 7.6, 7.7 y 7.8).

Lista de símbolos y abreviaciones

Abreviaciones	Descripciones	Definición
GECODE	entorno de desarrollo restricción genérica	page 18
TEU	veinte pies de unidades equivalentes	page 12
FEU	cuarentea pies de unidades equivalentes	page 12
MIT	Instituto Teconologico de Massachusetts	page 18
IP	Programación entera	page 19
CP	Programación por restricciones	page 16
CSP	Problema de satisfacción de restricciones	page 16
CSPADL	Problema de estibar de contenedores sobre loca- ciones por encima de cubierta	page 59
ISO	Organización Internacional de Normalización	page 17

Índice de figuras

1.1. Modelo total de planeación de estiba. [14][8]	6
2.1. Dimensiones comunes de un contenedor.	11
2.2. Imagen de un buque portacontenedores.	12
2.3. (a) Vista frontal de una bahía. (b) Vista lateral de una pila con carga parcial. Cada enchufe de poder indica si puede ir un contenedor refrigerado.	13
2.4. Los contenedores de color verde se descargarán antes que los rojos. (a) Pila sobre-estibada . (b) Escotilla sobre-estibada.	16
5.1. Índices de espacios (Slots)	31
5.2. Indexación de espacios en el buque	32
5.3. Coordenadas para cálculo de centro de gravedad	37
6.1. Índices por pila específica	47
6.2. Índices por pila general	48
6.3. Índices por niveles	49
7.1. Índices por pila	56
7.2. Índices por niveles	57
7.3. Índices por niveles y contenedores reales	57
8.1. Archivo de entrada.	61
8.2. Archivo de salida.	62
9.1. Tiempo de soluciones óptimas. Las siglas (M.O.) y (M.E.) hacen referencia al modelo original y al modelo extendido.	66
9.2. Diagramas de dispersión estrategia No.1 - Modelo original	68
9.3. Diagramas de dispersión estrategia No.2 - Modelo original	68
9.4. Diagramas de dispersión estrategia No.3 - Modelo original	68
9.5. Diagramas de dispersión estrategia No.1 - Modelo extendido	69

9.6. Diagramas de dispersión estrategia No.2 - Modelo extendido	69
9.7. Diagramas de dispersión estrategia No.3 - Modelo extendido	70

Índice de cuadros

3.1. Tabla de constantes.	22
3.2. Tabla de variables de decisión.	23
5.1. Tabla de constantes.	36
6.1. Tabla de variables de decisión.	40
6.2. Tabla de variables de decisión utilizadas para calcular el costo de la solución.	41
7.1. Tabla de constantes adicionales.	52
7.2. Tabla de variables de decisión.	53
9.1. Tabla de clasificación de instancias	64
9.2. Tabla de porcentajes de éxito	65

Bibliografía

- [1] Daniela Ambrosino, Davide Anghinolfi, Massimo Paolucci, and Anna Sciomachen. A new three-step heuristic for the master bay plan problem. *Maritime Economics & Logistics*, 11(1):98–120, 2009.
- [2] Daniela Ambrosino and Anna Sciomachen. A constraint satisfaction approach for master bay plans. *Water studies series*, pages 175–184, 1998.
- [3] Anastasios Haralampos Aslidis. *Optimal container loading*. PhD thesis, Massachusetts Institute of Technology, 1984.
- [4] Mordecai Avriel, Michal Penn, Naomi Shpirer, and Smadar Witteboon. Stowage planning for container ships to reduce the number of shifts. *Annals of Operations Research*, 76:55–71, 1998.
- [5] RC Botter and MA Brinati. Stowage container planning: A model for getting an optimal solution. 1991.
- [6] Lucas Antón De Luca, Nicolas German & Neller. Introducción a la ingeniería naval - buques portacontenedores.
- [7] Alberto Delgado. Models and algorithms for container vessel stowage optimization. 2013.
- [8] Alberto Delgado, Rune Møller Jensen, Kira Janstrup, Trine Høyer Rose, and Kent Høj Andersen. A constraint programming model for fast optimal stowage of container vessel bays. *European Journal of Operational Research*, 220(1):251–261, 2012.
- [9] Opher Dubrovsky, Gregory Levitin, and Michal Penn. A genetic algorithm with a compact solution encoding for the container ship stowage problem. *Journal of Heuristics*, 8(6):585–599, 2002.
- [10] efficient constraint solving toolkit GECODE An open, free. Definición del lenguaje gecode.
- [11] Jun-Gyu Kang and Yeong-Dae Kim. Stowage planning in maritime container transportation. *Journal of the Operational Research Society*, 53(4):415–426, 2002.

- [12] Feng Li, Chunhua Tian, Rongzeng Cao, and Wei Ding. An integer linear programming for container stowage problem. In *Computational Science-ICCS 2008*, pages 853–862. Springer, 2008.
- [13] Malcolm Yoke Hean Low, Xiantao Xiao, Fan Liu, Shell Ying Huang, Wen Jing Hsu, and Zhengping Li. An automated stowage planning system for large container ships.
- [14] Dario Pacino, Alberto Delgado, Rune Møller Jensen, and Tom Bebbington. Fast generation of near-optimal plans for eco-efficient stowage of large container vessels. In *Computational Logistics*, pages 286–301. Springer, 2011.
- [15] Amaya S. Definición del lenguaje c++.
- [16] Anna Sciomachen and Elena Tanfani. The master bay plan problem: a solution method based on its connection to the three-dimensional bin packing problem. *IMA Journal of Management Mathematics*, 14(3):251–269, 2003.
- [17] Zhang Wei-Ying, Lin Yan, and Ji Zhuo-Shang. Model and algorithm for container ship stowage planning based on bin-packing problem. *Journal of Marine Science and Application*, 4(3):30–36, 2005.
- [18] enciclopedia informatica Wikipedia. Definición de estiba.
- [19] ID Wilson and PA Roach. Container stowage planning: a methodology for generating computerised solutions. *Journal of the Operational Research Society*, pages 1248–1255, 2000.